TRANSIENT HEAT TRANSFER IN POROUS MEDIA

DANG DINH HIEP

TRANSIENT HEAT TRANSFER

IN POROUS MEDIA

* * * * *

Dang Dinh Hiep

TRANSIENT HEAT TRANSFER

IN POROUS MEDIA

by

Dang Dinh Hiep

Lieutenant, Vietnamese Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

United States Naval Postgraduate School
Monterey, California

1 9 6 5

TRANSIENT HEAT TRANSFER

IN POROUS MEDIA

by

Dang Dinh Hiep

This work is accepted as fulfilling
the thesis requirements for the degree of

MASTER OF SCIENCE

IN

MECHANICAL ENGINEERING

from the

United States Naval Postgraduate School

ABSTRACT

The general differential equations describing unsteady-state heat transfer with a fluid flowing through a porous medium are derived. These equations represent a physical model for heat transfer in thermal oil-recovery process, packed-bed chemical reactors, and heat regenerators. Fluid-solid convective heat transfer and longitudinal conduction in both the fluid and solid phases are considered. Laplace transformation and numerical inversion are used to solve the system of partial differential equations. A digital computer program obtains the numerical results which are compared to those of Green and Perry using finite difference technique, and to experimental data of Preston. Also presented are analytical solutions for the cases where the longitudinal conduction is neglected and the convective heat transfer coefficient is assumed to be infinite. These solutions are programmed and results are compared to those from the general case. The effect of different heat transfer mechanisms on temperature profiles at low fluid velocities is studied. The results show that this numerical method gives accurate results with relatively short computational time.

ACKNOWLEDGEMENTS

I would like to thank my thesis advisor, Dr. U. R. Kodres, Associate Professor, Department of Mathematics and Mechanics, USNPGS, for his constant guidance and help in the preparation of this thesis.

I also wish to express my appreciation to Dr. P. F. Pucci, Associate Professor, Department of Mechanical Engineering, USNPGS for his helpful suggestions and to Dr. J. E. Brock, Professor, Department of Mechanical Engineering, USNPGS for his review of this work.

Thanks are also due to Professor Don W. Green, Department of Chemical and Petroleum Engineering, University of Kansas, for providing technical information.

TABLE OF CONTENTS

Section	Title	Page
	Abstract	ii
	Acknowledgements	iii
	List of Illustrations	v
	Nomenclature	vi
1.	Introduction	1
2.	Literature Survey	4
3.	Mathematical Analysis	9
4.	Computer Programming	19
5.	Numerical Results	21
6.	Discussion of Results	28
7.	Conclusions	32
8.	Recommendations for Future Studies	33
9.	Bibliography	35
Appendix I	Analytical solution for zero longitudinal conduction in both the fluid and solid	37
Appendix II	Analytical solution for infinite heat transfer coefficient	44
Appendix III	General solution applied to a model of finite length	49
Appendix IV	Program listings	60

LIST OF ILLUSTRATIONS

Figure		Page
1.	Fluid-solid temperature differences; effect of dimensionless parameter λ .	21
2.	Comparison of generalized numerical solutions to simplified analytical solutions; dependence on λ .	22
3.	Comparison of generalized numerical solutions to simplified analytical solutions; dependence on k_s .	23
4.	Fluid temperature profiles; effect of λ .	24
5.	Fluid temperature profiles; effect of k_s .	25
6.	Comparison of numerical temperature profile to experimental data of Preston; system Copper-Water.	26
7.	Comparison of numerical temperature profile to experimental data of Preston; system Glass-Iso-Octane.	27

NOMENCLATURE

Symbol		Quantity	Unit
a	=	Surface area of solid particle per unit of bulk volume	ft^2/ft^3
A	=	Total heat transfer area	ft^2
A_f	=	Fluid cross sectional flow area	"
A_s	=	Matrix cross sectional flow area	"
c_f	=	Fluid phase specific heat	Btu/lb.^oF
c_s	=	Solid phase specific heat	"
C_1	=	$c_f \phi$	"
C_3	=	$c_f \phi + c_s(1-\phi)$	"
d_p	=	Average particle diameter	ft
F	=	Number of time units per hour	UT/hr
h	=	Heat transfer coefficient	Btu/hr ft ^oF
k_e	=	Effective thermal conductivity of porous medium, assuming solid and fluid temperature are equal	Btu/hr ft^oF/ft
k_s	=	Effective thermal conductivity of solid phase	"
k_s'	=	$k_s(1-\phi)$	Btu/hr ft^2 ^oF/ft
k_f	=	Effective thermal conductivity of fluid phase	"
k_f'	=	$k_f \phi$	"
k_{fc}	=	Molecular thermal conductivity of fluid	"
k_{fm}	=	Effective conductivity of fluid phase due only to fluid mixing or dispersion in porous medium	"
k_e^o	=	Static effective thermal conductivity of porous medium	"
L	=	Length of packed bed	"
s	=	Laplace transform variable	dimensionless

Symbol		Quantity	Unit
t	=	Dimensionless time parameter, $\frac{ha\theta}{W_s c_s}$	dimensionless
T_f	=	Fluid temperature	^oF
T_i	=	Injected fluid temperature	^oF
T_s	=	Solid temperature	^oF
u	=	Solid temperature fraction, T_s/T_i	dimensionless
v	=	Fluid temperature fraction, T_s/T_i	"
V_f	=	Fluid interstitial velocity	ft/hr
UT	=	Time unit	fraction of hr
x	=	Distance from point of fluid injection	ft
X	=	Dimensionless distance, $\frac{x}{L}$	dimensionless
Y	=	Dimensionless distance, $\left(\frac{ha}{k'_f}\right)^{\frac{1}{2}} x$	"
$\dot{\omega}_f$	=	Fluid mass flow rate	lb_m/hr
W_s	=	Mass of solid matrix	lb_m
N_{re}	=	Modified Reynolds number, $\frac{V_f d_p \rho_f}{\mu}$	dimensionless
N_{pe}	=	Peclet number $\frac{V_f d_p}{\alpha}$	"
NTU	=	Number of heat transfer units $\frac{hA}{\dot{\omega}_f c_f}$	"

GREEK LETTERS

α	=	Thermal diffusivity, $\frac{k}{\rho c}$	ft^2/hr
β	=	Ratio of thermal diffusivities, $\frac{\alpha_s}{\alpha_f}$	dimensionless
β'	=	$\frac{1}{\beta}$	"
γ	=	Ratio of thermal conductivities, $\frac{k'_f}{k'_s}$	"
λ	=	Dimensionless conduction parameter, $\left(\frac{ha}{k'_f}\right)^{\frac{1}{2}} \frac{\alpha_f}{V_f}$	"
λ'	=	Dimensionless conduction parameter, $\frac{k_s A_s}{\dot{\omega}_f c_f L}$	"
τ	=	Dimensionless time, $\left(\frac{ha}{k'_f}\right)^{\frac{1}{2}} V_f \theta$	"
θ	=	Time	hr

Symbol		Quantity	Unit
μ	=	Viscosity	lb_m/ft hr
ρ	=	Density	lb_m/ft^3
ϕ	=	Porosity of porous medium	dimensionless
ψ	=	Ratio of heat capacities per unit length	"

NOTE: An occasional term may appear in the body of the text that does not appear in this list. Such terms are used only once and are defined as they appear.

1. Introduction.

Thermal recovery operations are rapidly growing in importance throughout the oil producing industry. Large volumes of oil previously considered uneconomical to recover are being produced by thermal processes. The intense interest in the application of the thermal energy to oil reservoirs as a means of increasing the percentage of oil recovery has stimulated the research on the problem of heat transfer in porous media. Thermal recovery has seemed most applicable to reservoirs that contain very viscous oil at reservoir conditions. This is due primarily to two factors: the low recovery from viscous oil reservoirs by primary production or conventional secondary recovery methods and the significant reduction in viscosity that takes place when viscous oil is heated.

In these thermal methods, heat is injected or generated in the reservoir. The heated oil has its viscosity decreased thus making the removal from the reservoir easier. Thermal energy may be applied to a reservoir in different ways. The simplest processes are steam injection and hot water injection. In more complicated processes, the crude oil is burned at one end of the reservoir, forming a combustion zone which moves toward the other end. The product of combustion is a mixture of oil and condensed water, resulting from thermal cracking. No matter which method is used, the effect of heat on the production of oil and water should be known. Thus, a knowledge of various heat transfer mechanisms with their individual effects is required and also the temperature history at each point of the reservoir and the temperature distribution throughout the reservoir should be known.

Previous studies on heat transfer in porous media may be classified

into two groups. The first group considered that the main heat transfer mechanism involved in this problem is convection from the fluid surface to the solid surface. Thus the longitudinal conduction in both the fluid and solid phases are neglected. This case was intensely studied by Anzelius [1], Hausen [10], Nusselt [22], Schumann [29], and others. The second group assumed that the film resistance is negligible and that the heat is transferred solely by longitudinal conduction. This attack on the problem was made by Jenkins and Aronofsky [14]. Preston [24] used their solution to compare with the results from his experimental work. Authors on the problem of heat regenerators considered the matrix of the heat exchanger as a porous medium through which a gas is pumped. In this case the stored energy and the longitudinal conduction in the fluid were neglected. Green and Perry [7] have investigated the general case where both conduction in the direction of flow and convection from fluid to solid were considered in the mechanism of heat transfer. They used finite difference techniques to solve the general set of partial differential equations. This forward difference approach has its disadvantages because of the small time and space increments necessary.

It was the purpose of this thesis to use Laplace transforms to solve the differential equations derived from the heat balance for the general case. A FORTRAN program was set up for use with a CDC 1604 digital computer. By using the parametric values of Green and Perry [7] and of Preston [24], numerical solutions were obtained and checked against their solutions. Also, analytical solution to the differential equations of Schumann and Jenkins-Aronofsky were derived by using Laplace transforms. Two programs were set up and numerical solutions

were compared with the solutions to the general set of equations. The purpose of this comparison was to determine the relative importance of the different heat transfer mechanisms and how these mechanisms are affected by changes in the significant parameters.

The general case studied in Section 3 is limited to a model of infinite length. The mathematical derivations for the case of heat regenerators and of packed beds of finite length are presented in Appendix III. An outline is presented here of additional work which would be required to produce numerical results for this case.

2. <u>Literature Survey</u>

The theoretical solution to the problem of transient heating of porous media should provide:

a. The temperature history at a point in the porous medium as a function of time.

b. The temperature distribution throughout the length of the medium at a given time.

The mathematical and physical model of an oil reservoir is similar to that of a packed-bed or of a heat regenerator. Many theoretical studies have been made in these areas. The reservoir can be considered as a semi-infinite porous body through which the fluid is flowing. The following assumptions are usually made:

a. The initial solid and fluid temperature are equal throughout the length of the body.

b. The fluid is injected at one end. At time zero, its temperature is suddenly changed to a higher value and kept constant at this end.

c. The rate of fluid flow is constant.

d. The physical properties of fluid and solid are independent of temperature.

e. No temperature gradient exists in the direction perpendicular to the flow direction, i.e., the conductivity of the solid is infinite in that direction.

The basic mechanisms of heat transfer in a porous medium through which the fluid is flowing are:

(1) Storage of heat in an element of fluid.

(2) Conduction of heat through the solid and the fluid phases.

(3) Convection between the solid and fluid phases.

(4) radiation.

Radiation may play a significant role in the energy transfer encountered in the problem of transpiration of fluid in chemical reactors, heat shields and solar heat collectors. This mechanism is assumed negligible in an idealized model of a thermal oil recovery process, packed-bed chemical reactors and heat regenerators. The differential equations applied to the general case where both conduction and convection are considered can be derived from heat balance as presented in the next section. The original equations are:

<u>For the fluid phase</u>:

$$\rho_f c_f \phi \frac{\partial T_f}{\partial \theta} = - V_f \rho_f c_f \phi \frac{\partial T_f}{\partial x} + k_f \phi \frac{\partial^2 T_f}{\partial x^2} - ha(T_f - T_s) \qquad (1)$$

<u>For the solid phase</u>:

$$\rho_s c_s (1 - \phi) \frac{\partial T_s}{\partial \theta} = k_s (1 - \phi) \frac{\partial^2 T_s}{\partial x^2} + ha(T_f - T_s) \qquad (2)$$

where:

Subscript f refers to fluid phase.

s refers to solid phase.

T = temperature above a base temperature which is the initial bed temperature, ^{o}F

a = surface area of solid particle per unit of bulk volume, ft^2/ft^3

c = specific heat, Btu/lb.^{o}F

h = heat transfer coefficient, Btu/hr.ft^2.^{o}F

k = pseudo-thermal conductivity, Btu/hr.ft^2.^{o}F/ft

x = distance from point of fluid injection, ft

V_f = linear velocity of fluid, ft/hr

ϕ = bed porosity, dimensionless

θ = time, hours

ρ = density, lb/ft^3

The different mechanisms of heat transfer involved in the general heat balance equations were discussed in detail by Hadidi [9] .

Since an analytical solution to this set of differential equations is obviously difficult, all previous studies were confined to special cases, where either conduction or convection is neglected. An outline of this literature might be helpful to the reader.

Case 1: $k = 0, \quad 0 < ha < \infty$

By assuming that conduction in both phases is negligible, one can reduce the equations (1) and (2) to:

$$\rho_f c_f \phi \frac{\partial T_f}{\partial \theta} = - V_f \rho_f c_f \phi \frac{\partial T_f}{\partial x} - ha(T_f - T_s) \tag{3}$$

$$\rho_s c_s (1-\phi) \frac{\partial T_s}{\partial \theta} = ha(T_f - T_s) \tag{4}$$

This case was handled by Anzelius [1] , Schumann [29] , Nusselt [22] , Hausen [10], etc. Several techniques have been developed to solve the system of equations (3) and (4). C. E. Iliffe [12] presented an alternative method of solution to the same equations in a thermal analysis of the counterflow regenerative heat exchanger. Nahavandi and Weinstein [21] used Laplace transform and power series expansion to present a solution to the rotary heat exchanger problem. Lambertson [12] X and, recently, A. J. Willmott [31] presented a digital computer simulation of a thermal regenerator by using finite differences to solve this problem.

In Appendix I the writer derived the solution to this special case by simply using Laplace transforms. The same dimensionless parameters in the general case were used and the solution was then programmed to provide numerical data which were compared with the solution to the general problem. An alternate attack to the problem was made by Creswick [5] . In his analysis, he neglected the term $\rho_f c_f \phi \frac{\partial T_f}{\partial \theta}$

in equation (3) describing the heat gained by an element of the moving fluid, but he considered important the effect of longitudinal conduction in the solid by adding the term $k_s(1-\phi)\frac{\partial^2 T_s}{\partial x^2}$ to equation (4). These two equations were solved by finite difference techniques. Bahnke [2] used Creswick's equations and finite differences to solve for the conduction effect on effectiveness of the rotary regenerator. Recently, Moreland [20] applied Laplace transform and Gaussian quadrature for numerical inversion to get the solution to the "single blow" problem, using the same set of equations.

Case 2: $0 < k < \infty$; $ha = \infty$.

In this case, the fluid-solid boundary resistance was negligible, i.e., ha is infinite or $T_f = T_s$. But the porous body is considered as a homogeneous unit with longitudinal conduction in the direction of flow.

By substituting for the term ha $(T_f - T_s)$ in equation (1) its value derived from equation (2) and letting $T_f=T_s=T$, we have a combined equation:

$$\left[\rho_s c_s (1-\phi) + \rho_f c_f \phi\right] \frac{\partial T}{\partial \theta} = - V_f \rho_f c_f \phi \frac{\partial T}{\partial x} + k_e \frac{\partial^2 T}{\partial x^2} \tag{5}$$

where $k_e = k_f \phi + k_s(1-\phi)$ is the effective thermal conductivity of the porous medium. This approach to the problem of heating of porous media was offered by Jenkins and Aronofsky [14]. The writer's solution to equation (5) was derived by using Laplace transforms and is presented in Appendix II. A program was set up to provide numerical results which were checked against the solutions to the general case.

Jenkins and Aronofsky, after investigating the results and checking them against published data, mentioned that by selecting a value of k_e which gives the best agreement between experimental temperature profile and the analytical solution to equation (5), one can determine the combined dynamic thermal conductivity of the porous system.

Preston [24] in his experimental work, measured the static thermal conductivity of the porous system under no-flow conditions and the dynamic thermal conductivity under flow condition. He concluded that at velocities less than 0.05 ft/hr, (i.e., at velocities characteristic of flow in petroleum reservoirs), the effective thermal conductivity would equal the static thermal conductivity. For greater velocities, he stated that the effective thermal conductivity under flow condition increased with velocity. He concluded that at low rates of flow, the main mechanism of heat transfer in porous media could be assumed to be longitudinal conduction alone, i.e., the convection heat transfer could be neglected.

Case 3: Both k and ha are finite

This is the most general case where both conduction and convection are assumed to be important. The general set of differential equations given by equations (1) and (2) is too complicated for an analytical solution. In a recent paper, Green and Perry [7] reported the results obtained from a numerical analysis of the problem. They reduced these equations to difference equations of the forward difference type and solved for several values of parameters on an IBM 650 digital computer. Their solutions were checked against the results of Preston [24] with close agreement. The writer's approach to solve the system of differential equations (1) and (2) is presented in the next section. A computer program was set up for use with a CDC 1604 computer. Numerical solutions were compared with the results of Green and Perry [7] and of Preston [24] .

3. Mathematical Analysis

The differential equations (1) and (2) can be derived from heat balance as follows:

a. For the fluid phase:

Heat stored in an element of fluid $= \rho_f c_f \phi \frac{\partial T_f}{\partial \theta}$

Convection by moving fluid $= \rho_f V_f c_f \phi \frac{\partial T_f}{\partial x}$

Conduction in the fluid $= k_f \phi \frac{\partial^2 T_f}{\partial x^2}$

Heat transfer to the fluid element by convection $= ha(T_f - T_s)$

The heat balance yields:

$$\rho_f c_f \phi \frac{\partial T_f}{\partial \theta} = -\rho_f c_f V_f \phi \frac{\partial T_f}{\partial x} + k_f \phi \frac{\partial^2 T_f}{\partial x^2} - ha(T_f - T_s) \qquad (6)$$

This is the same as equation (1).

b. For the solid phase:

Heat gained by an element of solid $= \rho_s c_s (1-\phi) \frac{\partial T_s}{\partial \theta}$

Heat transferred to the solid element by convection $= ha(T_f - T_s)$

Heat transferred by conduction from the solid element $= k_s (1-\phi) \frac{\partial^2 T_s}{\partial x^2}$

The heat balance gives:

$$\rho_s c_s (1-\phi) \frac{\partial T_s}{\partial \theta} = k_s (1-\phi) \frac{\partial^2 T_s}{\partial x^2} + ha(T_f - T_s) \qquad (7)$$

This is the same as equation (2)

Let us define two new dimensionless variables:

y = dimensionless distance $= \left(\frac{ha}{k_f \phi}\right)^{\frac{1}{2}} x$

τ = dimensionless time $= \left(\frac{ha}{k_f \phi}\right)^{\frac{1}{2}} V_f \theta$

Substituting these new variables into (6), we get:

$$\frac{\partial T_f}{\partial \tau} = -\frac{\partial T_f}{\partial y} + \left(\frac{ha}{k_f \phi}\right)^{\frac{1}{2}} \left(\frac{k_f}{\rho_f c_f V_f}\right) \frac{\partial^2 T_f}{\partial y^2} - \left(\frac{ha}{k_f \phi}\right)^{\frac{1}{2}} \left(\frac{k_f}{\rho_f c_f V_f}\right) (T_f - T_s) \qquad (8)$$

We introduce the dimensionless parameter:

$$\lambda = \left(\frac{ha}{k_f \phi}\right)^{\frac{1}{2}} \frac{\alpha_f}{V_f} \qquad \text{where} \qquad \alpha_f = \frac{k_f}{\rho_f c_f}$$

Equation (8) becomes:

$$\frac{\partial T_f}{\partial \tau} = -\frac{\partial T_f}{\partial y} + \lambda \frac{\partial^2 T_f}{\partial y^2} - \lambda (T_f - T_s) \tag{9}$$

By the same substitution, equation (7) becomes:

$$\frac{\partial T_s}{\partial \tau} = \lambda \frac{\alpha_s}{\alpha_f} \frac{\partial^2 T_s}{\partial y^2} + \lambda \left(\frac{\alpha_s}{\alpha_f}\right)\left(\frac{k'_f}{k'_s}\right)(T_f - T_s) \tag{10}$$

where $\alpha_s = \frac{k_s}{\rho_s c_s}$

$$k'_f = k_f \phi$$

$$k'_s = k_s (1 - \phi)$$

Let $u = \frac{T_s}{T_i}$ and $v = \frac{T_f}{T_i}$

where T_i is the injected fluid temperature

$\beta = \frac{\alpha_s}{\alpha_f}$ and $\gamma = \frac{k'_f}{k'_s}$

the system of equations is then:

$$\frac{\partial u}{\partial \tau} = (\lambda \beta) \frac{\partial^2 u}{\partial y^2} + (\lambda \beta \gamma)(v - u) \tag{11}$$

$$\frac{\partial v}{\partial \tau} = \lambda \frac{\partial^2 v}{\partial y^2} - \frac{\partial v}{\partial y} - \lambda (v - u) \tag{12}$$

Boundary and initial conditions

(1) IC - The initial fluid and matrix temperatures are uniform and equal. The base scale temperature can be chosen so that these temperatures can be taken as zero for convenience:

$$u(y, 0) = v(y, 0) = 0$$

(2) BC - At y = o and $\tau = o^+$, the injected fluid temperature is suddenly changed to a different higher value and held constant thereafter. The input temperature is thus a step function:

$$v(0, \tau) = 1$$

(3) BC - At $y = 0$ and $\tau = 0^+$, the solid temperature is assumed to instantaneously rise to the value of the step input temperature of the fluid:

$$u(0,\tau) = 1$$

(4) BC - As y approaches infinity, for all τ , the fluid and solid temperatures decrease to their initial value:

$$u(\infty,\tau) = v(\infty,\tau) = 0$$

By calculating u from the equation (12), and differentiating it with respect to τ and y , we get the following equations:

$$u = \frac{1}{\lambda}\left[\frac{\partial v}{\partial \tau} - \lambda\frac{\partial^2 v}{\partial y^2} + \frac{\partial v}{\partial y} + \lambda v\right] \tag{13}$$

$$\frac{\partial u}{\partial \tau} = \frac{1}{\lambda}\frac{\partial^2 v}{\partial \tau} - \frac{\partial^3 v}{\partial y^2 \partial \tau} + \frac{1}{\lambda}\frac{\partial^2 v}{\partial y \partial \tau} + \frac{\partial v}{\partial \tau} \tag{14}$$

$$\frac{\partial^2 u}{\partial y} = \frac{1}{\lambda}\frac{\partial^3 v}{\partial y^2 \partial \tau} - \frac{\partial^4 v}{\partial y^4} + \frac{1}{\lambda}\frac{\partial^3 v}{\partial y^3} + \frac{\partial^2 v}{\partial y^2} \tag{15}$$

Substituting the terms u , $\frac{\partial u}{\partial \tau}$ and $\frac{\partial^2 u}{\partial y^2}$ into equation (11) and rearranging the terms yields the following differential equation in v :

$$(\lambda\beta)\frac{\partial^4 v}{\partial y^4} - \beta\frac{\partial^3 v}{\partial y^3} - (\beta+1)\frac{\partial^3 v}{\partial y^2 \partial \tau} - \beta\lambda(\gamma+1)\frac{\partial^2 v}{\partial y^2} +$$
$$+ (\beta\gamma)\frac{\partial v}{\partial y} + \frac{1}{\lambda}\frac{\partial^2 v}{\partial y \partial \tau} + (\beta\gamma+1)\frac{\partial v}{\partial \tau} + \frac{1}{\lambda}\frac{\partial^2 v}{\partial \tau} = 0 \tag{16}$$

Let $\bar{v}$ be the Laplace transform of v and s the transformed variable:

$$\bar{v}(y,s) = \mathcal{L}\left\{v(y,\tau)\right\}$$

The following formulas for Laplace transformation are used:

$$\mathcal{L}\left\{\frac{\partial^n v}{\partial y^n}\right\} = \frac{\partial^n \bar{v}}{\partial y^n}$$

$$\mathcal{L}\left\{\frac{\partial v}{\partial \tau}\right\} = s\bar{v} - v(y,0)$$

$$= s\bar{v} \quad \text{(from the initial condition)}$$

$$\mathcal{L}\left\{\frac{\partial^2 v}{\partial \tau}\right\} = s^2\bar{v} - s\,v(y,0) - \frac{\partial v}{\partial \tau}(y,0)$$

$$= s^2\bar{v}$$

$$\mathcal{L}\left\{\frac{\partial^2 v}{\partial y\,\partial \tau}\right\} = \frac{\partial}{\partial y}\left[\mathcal{L}\left\{\frac{\partial v}{\partial \tau}\right\}\right]$$

$$= \frac{\partial}{\partial y}\left[s\bar{v} - v(y,0)\right]$$

$$= s\,\frac{\partial \bar{v}}{\partial y}$$

$$\mathcal{L}\left\{\frac{\partial^3 v}{\partial y^2\,\partial \tau}\right\} = \frac{\partial}{\partial y}\left[\mathcal{L}\left\{\frac{\partial^2 v}{\partial y\,\partial \tau}\right\}\right]$$

$$= s\,\frac{\partial^2 \bar{v}}{\partial y}$$

Substituting for the partial derivatives in the equation (16) their Laplace transforms yields the following subsidiary equation:

$$\beta\lambda\,\frac{d^4\bar{v}}{dy^4} - \beta\,\frac{d^3\bar{v}}{dy^3} - \left[(\beta+1)s + \beta\lambda(\gamma+1)\right]\frac{d^2\bar{v}}{dy^2} + \left(\beta\gamma + \frac{s}{\lambda}\right)\frac{d\bar{v}}{dy} + \left[\frac{s^2}{\lambda} + (\beta\gamma+1)\,s\right]\bar{v} = 0 \tag{17}$$

This is an ordinary differential equation in $\bar{v}$. The corresponding auxiliary equation is:

$$(\beta\lambda)\,r^4 - \beta r^3 - \left[(\beta+1)\,s + \beta\lambda(\gamma+1)\right]r^2 + \left(\beta\gamma + \frac{s}{\lambda}\right)r + \left[\frac{s^2}{\lambda} + (\beta\gamma+1)\,s\right] = 0 \tag{18}$$

The general solution in the Laplace s plane for the fluid temperature is then:

$$\bar{v} = C_1(s)\,e^{r_1 y} + C_2(s)\,e^{r_2 y} + C_3(s)\,e^{r_3 y} + C_4(s)\,e^{r_4 y} \tag{19}$$

where r_1, r_2, r_3, r_4 are the roots of the equation (18). Since some coefficients of equation (18) are functions of s, which is a complex number, the roots r_1, r_2, r_3, r_4 are expected to be complex numbers.

The boundary conditions (2), (3) and (4) are transformed and then used to determine the constants C_n:

BC (2) : $\bar{v}(0,s) = \frac{1}{s}$

BC (3) : $\bar{u}(0,s) = \frac{1}{s}$

BC (4) : $\bar{u}(\infty,s) = \bar{v}(\infty,s) = 0$

Applying the BC (4), it is observed that $\bar{v}(\infty,s)$ may only be zero if the exponents in equation (19) are negative, i.e., if the real parts of the roots are negative. Since the parameters in the coefficients are unknown, the number of roots with negative real parts cannot be predicted by using Routh's criterion. Many computer test runs were made to investigate the behavior of the roots by using a wide range of parameters. Results from these tests showed that only two roots have negative real parts. The following derivations were based on that remark. If r_3 and r_4 are assumed to be the roots with positive real parts, then C_3 and C_4 in equation (19) must be zero and $\bar{v}$ is reduced to:

$$\bar{v} = C_1(s)\, e^{r_1 y} + C_2(s)\, e^{r_2 y} \tag{19a}$$

Applying the BC (2) to the above equation, one obtains:

$$C_1(s) + C_2(s) = \frac{1}{s} \tag{20}$$

Taking the partial derivatives of $\bar{v}(y,s)$ given by equation (19a) and evaluating them at y = 0 gives:

$$\frac{\partial \bar{v}}{\partial y}(0,s) = r_1 C_1(s) + r_2 C_2(s)$$

$$\frac{\partial^2 \bar{v}}{\partial y^2}(0,s) = r_1^2 C_1(s) + r_2^2 C_2(s)$$

We calculate the transform of equation (13):

$$\bar{u}(y,s) = -\frac{\partial^2 \bar{v}}{\partial y^2} + \frac{1}{\lambda}\frac{\partial \bar{v}}{\partial y} + \left(1 + \frac{s}{\lambda}\right)\bar{v} \tag{21}$$

Applying the BC (3) to the above equation and using the values of

$\frac{\partial \bar{v}}{\partial y}(0,s)$ and $\frac{\partial^2 \bar{v}}{\partial y^2}(0,s)$ above, we have:

$$\bar{u}(0,s) = \frac{1}{s} = -\sum_{n=1}^{2} r_n^2 C_n(s) + \frac{1}{\lambda}\sum_{n=1}^{2} r_n C_n(s) + \left(1 + \frac{s}{\lambda}\right)\frac{1}{s}$$

or

$$\sum_{n=1}^{2} r_n^2 C_n(s) - \frac{1}{\lambda}\sum_{n=1}^{2} r_n C_n(s) = \frac{1}{\lambda} \tag{22}$$

Let :

$$Z_n \equiv \left(r_n^2 - \frac{1}{\lambda} r_n\right) \qquad , n = 1, 2$$

then equation (22) may be written as:

$$\sum_{n=1}^{2} Z_n C_n(s) = \frac{1}{\lambda} \tag{23}$$

The functions C_1 and C_2 may be found from the matrix equation:

$$\begin{bmatrix} 1 & 1 \\ Z_1 & Z_2 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} \frac{1}{s} \\ \frac{1}{\lambda} \end{bmatrix}$$

It follows that:

$$C_1 = \frac{\frac{Z_2}{s} - \frac{1}{\lambda}}{Z_2 - Z_1}$$

$$C_2 = \frac{\frac{1}{\lambda} - \frac{Z_1}{s}}{Z_2 - Z_1}$$

Substituting C_1 and C_2 into equation (19) results in:

$$\bar{v}(y,s) = \frac{1}{Z_2 - Z_1}\left[\left(\frac{Z_2}{s} - \frac{1}{\lambda}\right)e^{r_1 y} + \left(\frac{1}{\lambda} - \frac{Z_1}{s}\right)e^{r_2 y}\right] \tag{24}$$

The value of $\bar{u}(y,s)$ can be given by equation (21):

$$\bar{u}(y,s) = -\frac{\partial^2 \bar{v}}{\partial y^2} + \frac{1}{\lambda}\frac{\partial \bar{v}}{\partial y} + \left(1 + \frac{s}{\lambda}\right)\bar{v}$$

$$= -\sum_{n=1}^{2} r_n^2 C_n e^{r_n y} + \frac{1}{\lambda}\sum_{n=1}^{2} r_n C_n e^{r_n y} + \left(1 + \frac{s}{\lambda}\right)\bar{v}$$

$$= \left(-r_1^2 + \frac{1}{\lambda} r_1\right) C_1 e^{r_1 y} + \left(-r_2^2 + \frac{1}{\lambda} r_2\right) C_2 e^{r_2 y} + \left(1 + \frac{s}{\lambda}\right)\bar{v} \tag{25}$$

To determine the values of the fluid temperature and solid temperature at any given time and at any point of the bed, the inverse transform of equation (24) and (25) should be found. This requires the evaluation of the inverse integral:

$$f(t) = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{st} F(s)\, ds$$

Where C is a real constant that exceeds the real part of each of the singular points of F(s) which is $\bar{v}(y,s)$ and $\bar{u}(y,s)$ in this case.

If the analytical form of the function $\bar{v}(y,s)$ were known and its poles and branch points could be located without difficulty, the inversion integral in (26) could be evaluated by a suitable deformation of the path of integration and the use of Cauchy's theorem on the residues. But in our case, $\bar{v}(y,s)$ and $\bar{u}(y,s)$ are functions of the roots of the quartic equation (19). These roots could be analytically calculated but their analytical forms would be so complicated that the evaluation of the inverse integral is hopeless. Thus, given numerical values of the parameters, $\bar{v}(y,s)$ and $\bar{u}(y,s)$ could only be evaluated as functions of s. Then some approximate numerical inversion scheme must be used.

The need for inverting Laplace transforms has been experienced in many fields, and approximate inversion methods have been developed in connection with several subjects. Thomas L. Cost [4] in applying numerical Laplace transform inversion to viscoelastic stress analysis, presented a unified treatment of the most promising approximate inversion methods in his paper. Among these, the orthogonal polynomial inversion methods of Papoulis [23] and of Lanczos [18] are mathematically well founded. Legendre and Laguerre orthogonal polynomials were used. Recently, Moreland [20] in his thesis on the "single blow"

problem, used the technique developed by H. Hurwitz [11] for numerical quadrature of Fourier transform integrals and adapted by L. S. Schmittroth [28] for the inversion of Laplace transforms. But the most sophisticated investigation on the numerical inversion method was made by Salzer [25]. In his early paper in 1955, he derived the properties of a certain set of orthogonal polynomials $P_n(x)$, that play a role in inversion integrals $\frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{st}F(s)\,ds$, similar to those of the Laguerre polynimials which are used to evaluate the direct Laplace transform integrals $\int_0^{\infty} e^{-st} f(t)\,dt$. A short table of weights and zeros was also furnished by the author. In his later paper [26], Salzer presented a table of weights which may be used in conjunction with values of F(s) evaluated at integral values of s, using the formula:

$$\frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{st}F(s)\,ds \cong \sum_{k=1}^{n} A_k^{(m)}(t)\,F(k)$$

This method is suitable for hand computers since both the weights and zeros are real numbers. In another paper on Laplace transforms [27], he presented an extensive table of complex zeros and Christoffel numbers up to order n = 16 and with 15 significant figures, for use with his first method. Since this is the most promising method, it has been chosen to invert the $\bar{v}(y,s)$ and $\bar{u}(y,s)$ given by equation (24) and (25). An outline of this method is presented here.

Salzer stated that if F(s) is really the Laplace transform of a function f(t), it must behave like a polynomial in the variable $\frac{1}{s}$ without a constant term along the line $c-i\infty$, $c+i\infty$. Then one may find f(t) numerically using new quadrature formulas similar to those employing the zeros of Laguerre polynomials in the direct L.T. or the

zeros of Legendre and Chebyscheff polynomials in the methods of Lanczos and Papoulis. Suitable choice of s_k yields an n-point quadrature formula that is exact when ρ_{2n} is any arbitrary polynomial of the $2n^{th}$ degree in $x = \frac{1}{s}$ without a constant term. Thus:

$$\frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^s \rho_{2n}\left(\frac{1}{s}\right) ds \cong \sum_{k=1}^{n} A_k^{(n)} \rho_{2n}\left(\frac{1}{s_k}\right) \tag{26}$$

In the above formula, $x_k = \frac{1}{s_k}$ are the zeros of the orthogonal polynomial $p_n(x) \cong \prod_{k=1}^{n}(x - x_k)$ derived from the generalized Bessel polynomial defined by H. L. Krall and O. Frink [16] as:

$$y_n(x, a, b) = \sum_{k=0}^{n}\binom{n}{k}(n+k+a-2)^{(k)}\left(\frac{x}{b}\right)^k$$

$P_n\left(\frac{1}{s}\right)$ is proved to be $(-1)^n e^{-s} s^n \frac{d^n}{ds^n}\left(\frac{e^s}{s^n}\right)$.

The orthogonal property is:

$$\frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} \frac{e^s}{s^k} P_n\left(\frac{1}{s}\right) ds = 0 \qquad k = 1,2,\ldots,n.$$

$A_k^{(n)}$ in formula (26) are Christoffel numbers.

There is no loss of generality if equation (26) is written as:

$$f(t) = \frac{1}{2\pi i t}\int_{c_1-i\infty}^{c_1+i\infty} e^u F\left(\frac{u}{t}\right) du = \sum_{k=1}^{n} A_k^{(n)} \rho_{2n}\left(\frac{1}{u_k^{(n)}}\right)$$

where $u = st$.

so that $F\left(\frac{u}{t}\right)$ is still a polynomial in $\frac{1}{u}$, without a constant term, if t is specified numerically. The roots $\frac{1}{u_k^{(n)}}$ and the Christoffel numbers $A_k^{(n)}$ are all complex, except when n is odd. They were provided in Salzer's paper [27].

Recently, N. Skobli͡a of the Academy of Sciences of USSR Moscow, has published a booklet [30] presenting tables for the numerical inversion

of Laplace transforms. This method is more general than Salzer's method. It enables one to evaluate:

$$f(t) = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} \frac{1}{s^m} e^{st} F(s)\,ds$$

where $c > 0$ and c lies to the right of all singularities of F(s) and m = 0.1(0.1)3.0. The case m = 1 has been treated by H. Salzer in his papers referred to above. The difference between these two methods is that Salzer's quadrature formula is exact if F(s) is a polynomial in $\frac{1}{s}$ of degree 2n such that $F(\infty) = 0$. In Skobli͡a's method, the quadrature formula is exact if F(s) is of degree (2n-1), but $F(\infty)$ need not vanish. Thus the Christoffel numbers in Salzer's table differ from those of Skobli͡a but the zeros are the same. Since these tables are not yet available at the USNPGS Library, a comparison of the results from two methods has not been possible.

4. Computer Programming

Preliminary test programs were set up to investigate the behavior of the roots of the auxiliary equation (18). Three Library subroutines solving the polynomials with complex coefficients were used. The subroutine ROOTS2 using MULLER's method proved to be unsuitable for this unusual equation. The roots did not converge after 25 iterations. Increasing the maximum number of iterations from 25 to 50 produced convergence but the computational time was too long as compared with the other subroutines. The subroutine COMSUB using Newton Raphson method was the fastest but for some range of data, it failed. Finally, the subroutine POLYRT using LEHMER's and NEWTON's method was tested. This subroutine gave satisfactory results although it was still slow. A combined program using both POLYRT and COMSUB was set up. Since the zeros of Salzer's polynomial do not largely vary from one to another, for each set of coefficients, the first run used POLYRT; the roots provided by that run are used as guessed roots in subroutine COMSUB. Each time the COMSUB fails the POLYRT is used again.

The subroutine VUBAR1 corresponds to the general case. It calculates the roots of equation (18), selects the roots with negative real parts, computes C_1 and C_2 and complex quantities $\bar{u}, \bar{v}$, then sends them back to the main program TEMFLUI1 with each set of zeros and Christoffel numbers corresponding to an order m. This enables the main program to provide a set of values of v and u . It was found that increasing the order of polynomials in $\left(\frac{1}{s_k}\right)$ did not necessarily increase the accuracy of v or u . Plotting the values of v vs the order m showed that v does not converge as m increases. On the contrary, the values oscillate at random. Attempting to choose an optimum m also failed. But all values of v agree to at least 3 significant figures.

Test runs for the limiting cases showed that the numerical results obtained by numerical inversion agree with the analytical solutions (provided by the programs Schumann and Jenkins) also up to 3 decimals. Finally, in order to shorten the computational time, it was decided that only the zeros and weights of order from 11 to 16 were used and the average of six values of v and u was taken. For each curve of v vs distance or time, only a limited number of points were calculated from numerical inversion. The intermediate points were interpolated by the subroutine AITKENF.

The analytical solutions for special cases in Schumann and Jenkins-Aronofsky problems were derived by the writer and presented in Appendicies I and II. Their numerical solutions were provided by programs SCHUMANN and JENKINS. Results from these programs were compared with the results of the general case to show the relative importance of various heat transfer mechanisms.

The mathematical derivations applied to the case of heat regenerators and of packed-bed of finite length are presented in Appendix III where the usual dimensionless parameters in heat regenerator problems were used. The problem turned out to be more complicated because all the roots of the characteristic equation must be used. Additional boundary conditions were used in order to be able to calculate all the four constants $C_i(s)$. A subroutine may be written for this case and numerical results may be compared with Creswick's results [5] .

Fig. 1 - Fluid-solid temperature differences; effect of dimensionless parameter λ

$$\frac{\alpha_s}{\alpha_f} = 3.17 \quad ; \quad \frac{k'_f}{k'_s} = 0.337$$

————	Fluid temperature
- - - - - - -	Solid temperature
⊙	Fluid temperature, Green and Perry
△	Solid temperature, Green and Perry

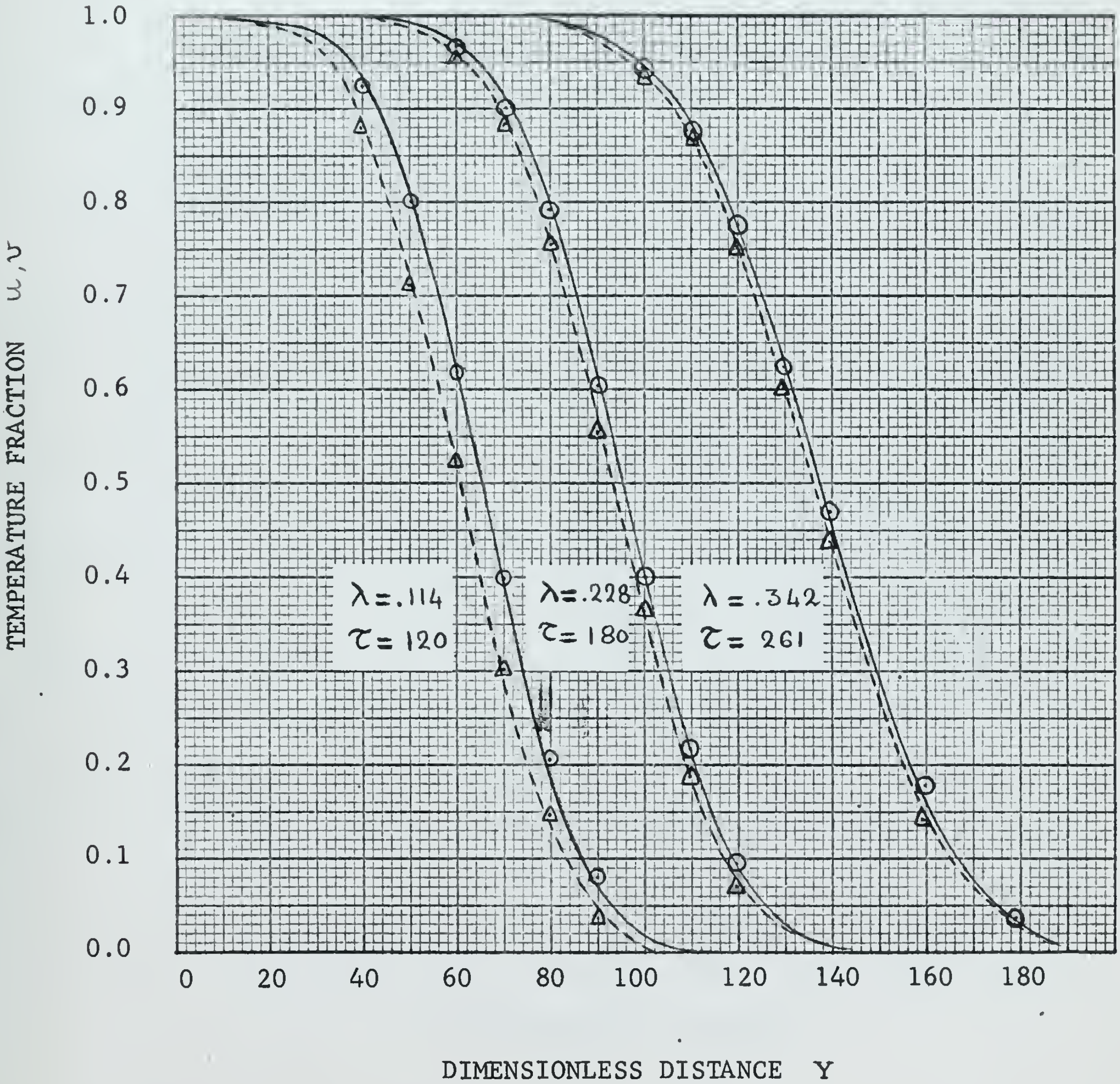


Figure 2. Comparison of generalized numerical solution to simplified analytical solution; effect of dimensionless parameter λ .

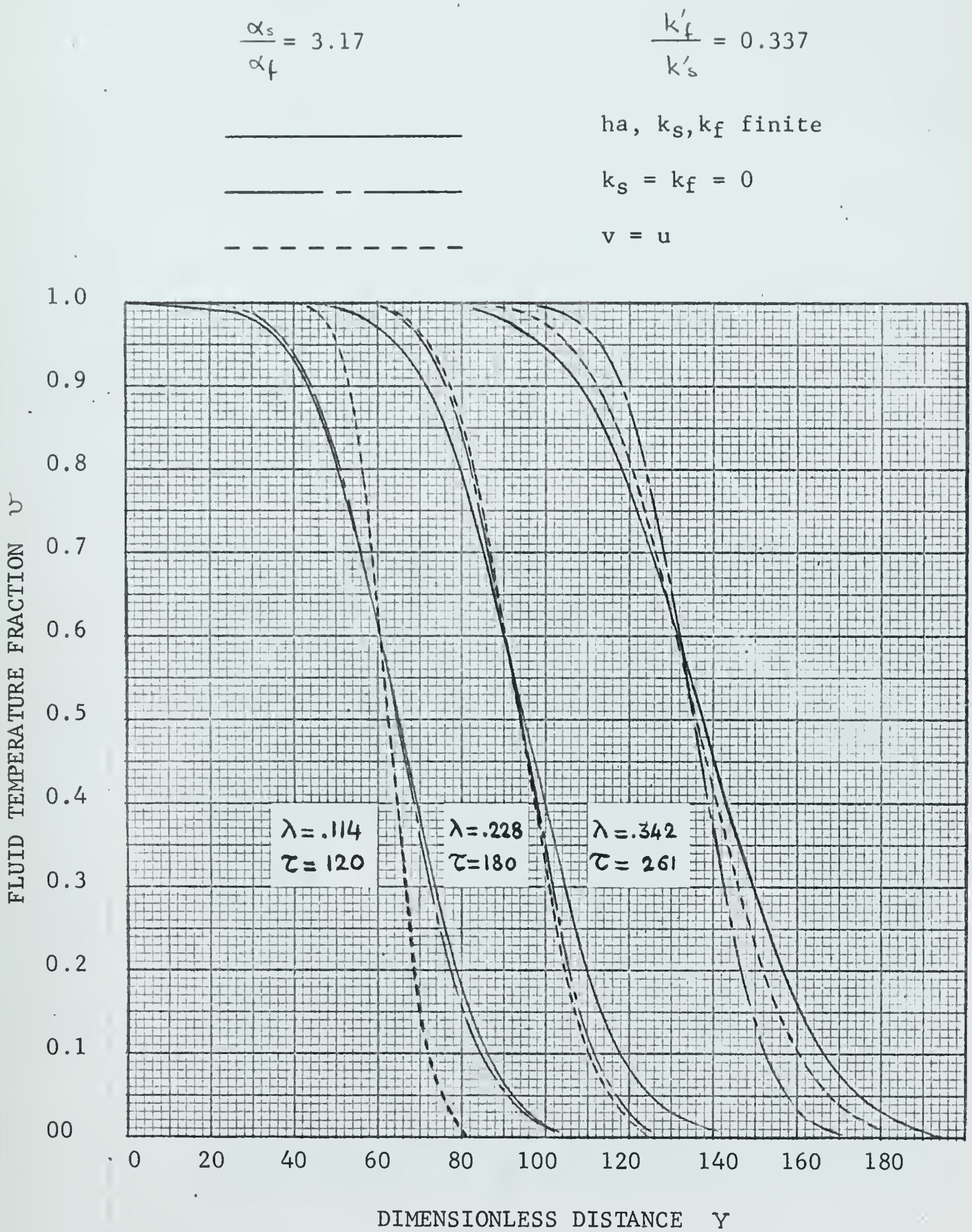


Figure 3. Comparison of generalized numerical solutions to simplified analytical solutions; effect of solid phase thermal conductivity.

	Curve A	Curve B
$\frac{\alpha_s}{\alpha_f}$	0.883	12.83
$\frac{k'_f}{k'_s}$	1.55	0.1065
τ	7.2	103.
λ	.342	.342

ha, k_f, k_s finite

$k_s = k_f = 0$

$T_f = T_s$

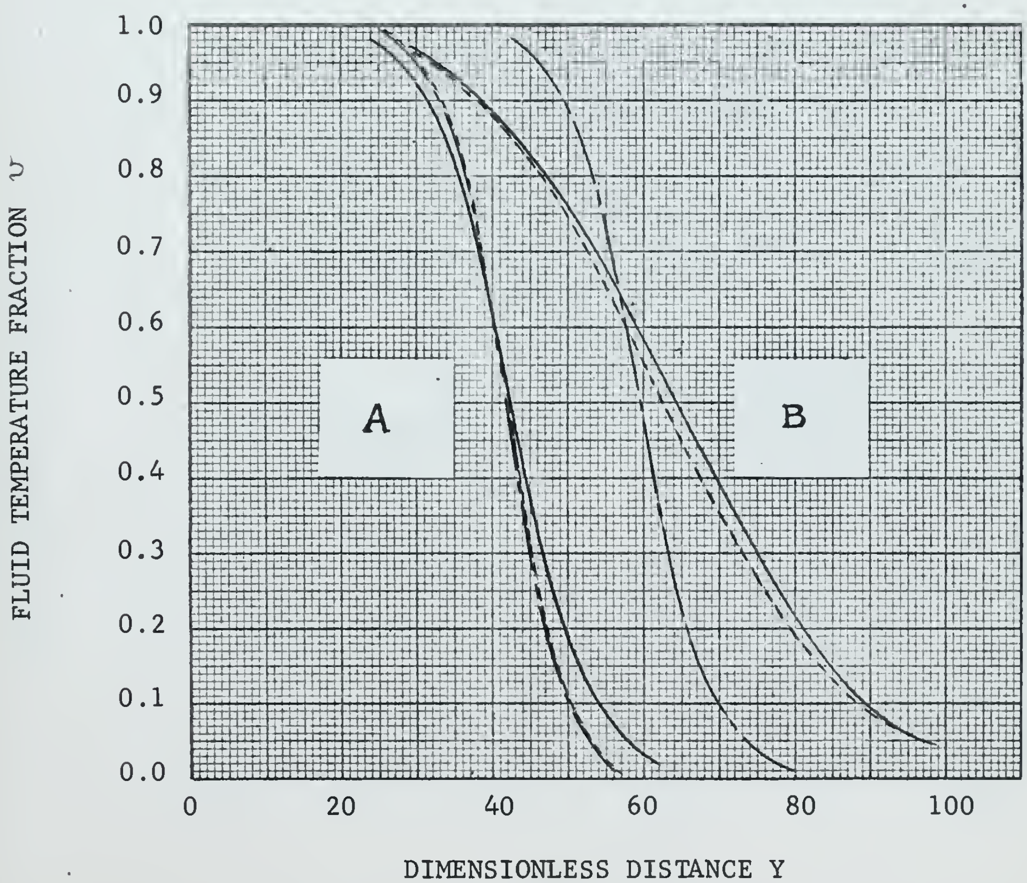


Figure 4. Fluid temperature profiles; effect of dimensionless parameter λ.

$$\frac{k'_f}{k_s} = 0.337$$

$$x = 1.0 \text{ ft}$$

$$V_f = 1.0 \text{ ft/hr}$$

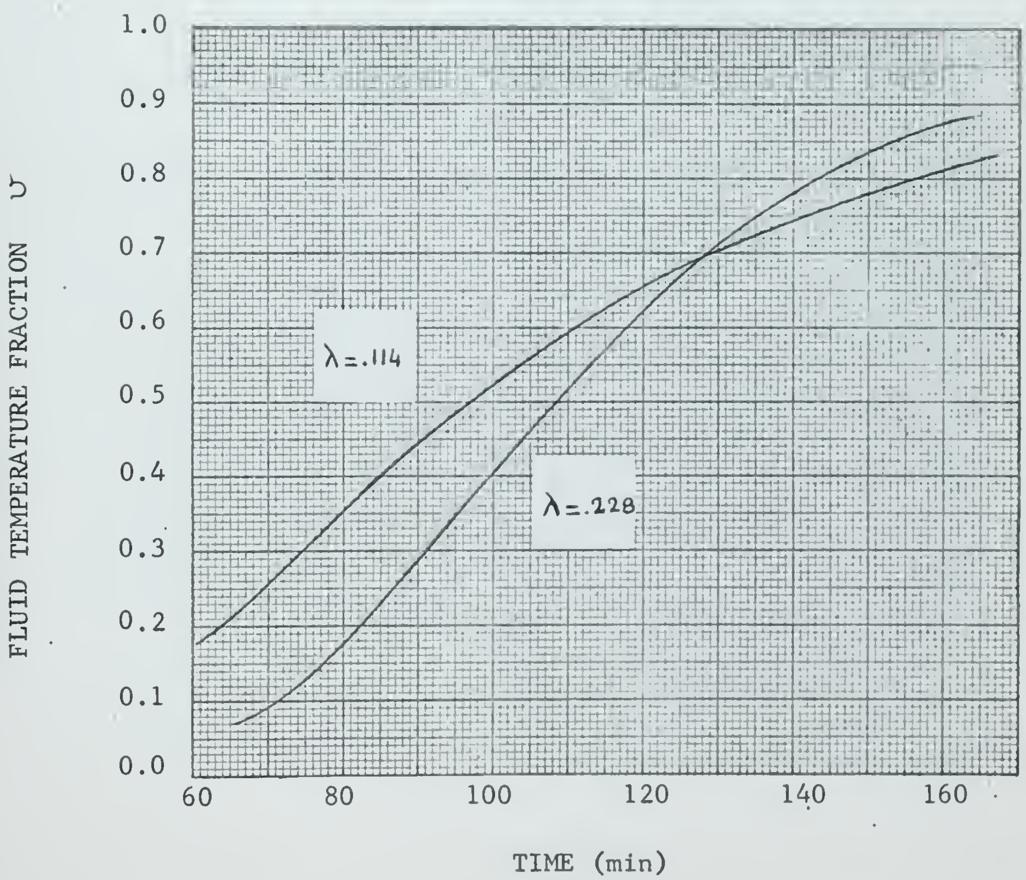


Figure 5. Fluid temperature profiles; effect of solid thermal conductivity.

$$\left(\frac{\alpha_s}{\alpha_f}\right)\left(\frac{k'_f}{k'_s}\right) = 1.37$$

$$\lambda = 0.342$$

$$x = 0.5 \text{ ft}$$

$$V_f = 1 \text{ ft/hr}$$

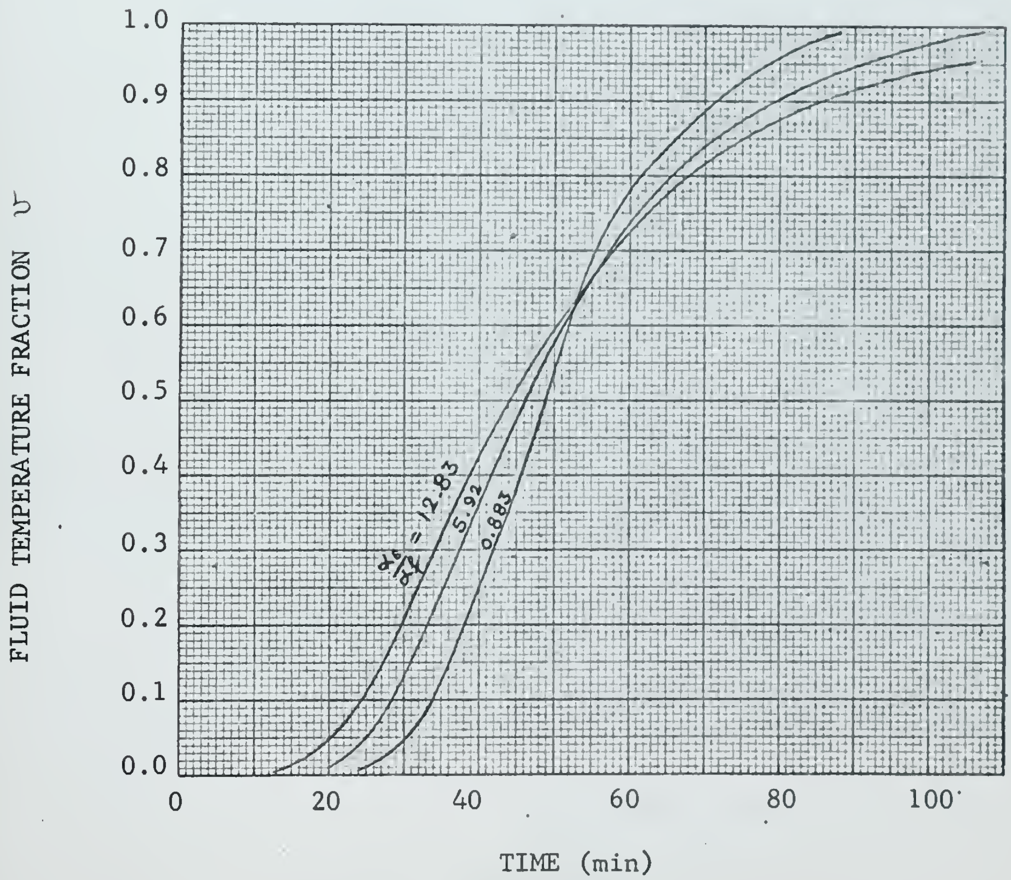


Figure 6. Comparison of numerical temperature profile to PRESTON'S experimental data.

POROUS SYSTEM : MESH OF COPPER-WATER

k_e^o	Btu/hr ft^2°F/ft	=	3.86
k_s	"	=	6.24
k_f	"	=	.461
ρ_s	lb/ft^3	=	556.3
ρ_f	"	=	61.9
c_s	Btu/lb°F	=	.0923
c_f	"	=	.9993
ϕ		=	.403
ha	Btu/ft^3°F hr	=	17300.
Vel	ft/hr	=	23.
X	ft	=	0.958
F	Time Unit/hr	=	360.

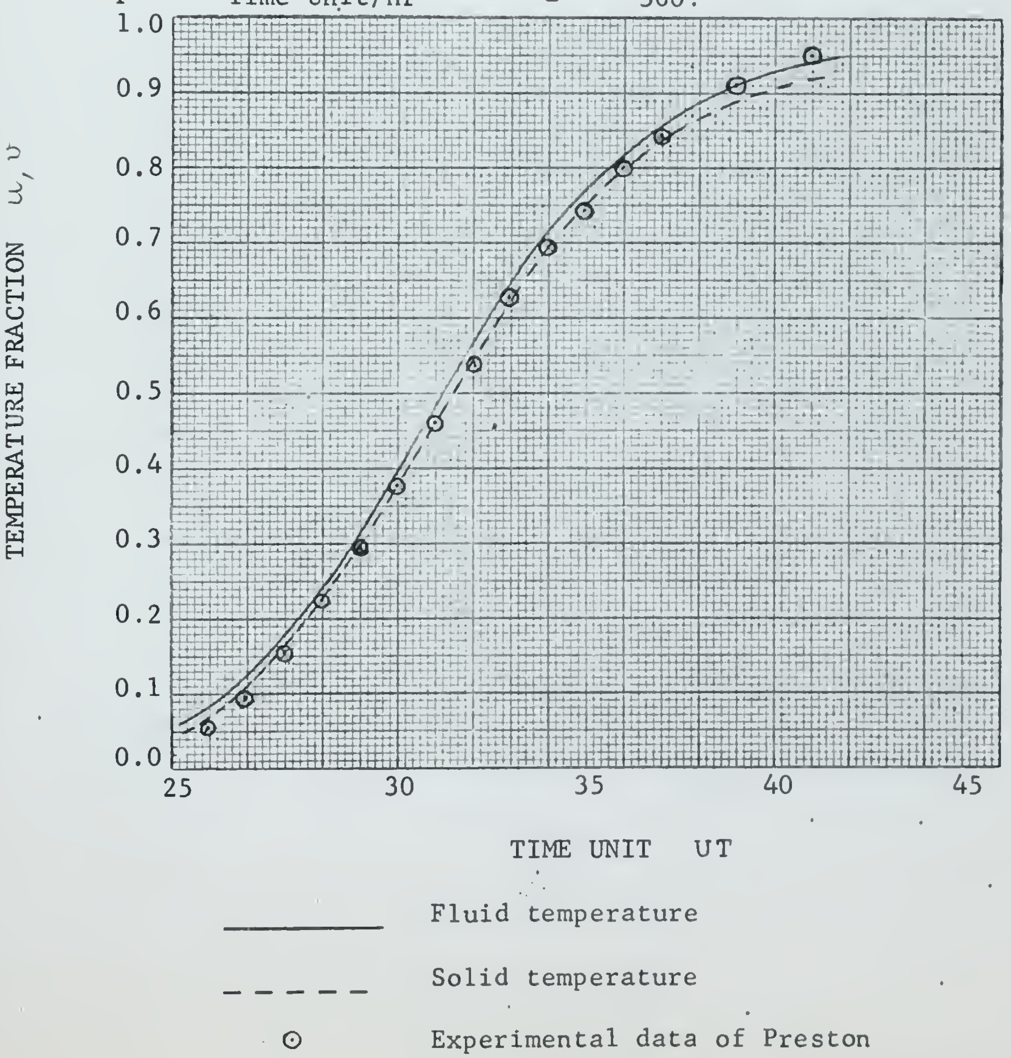
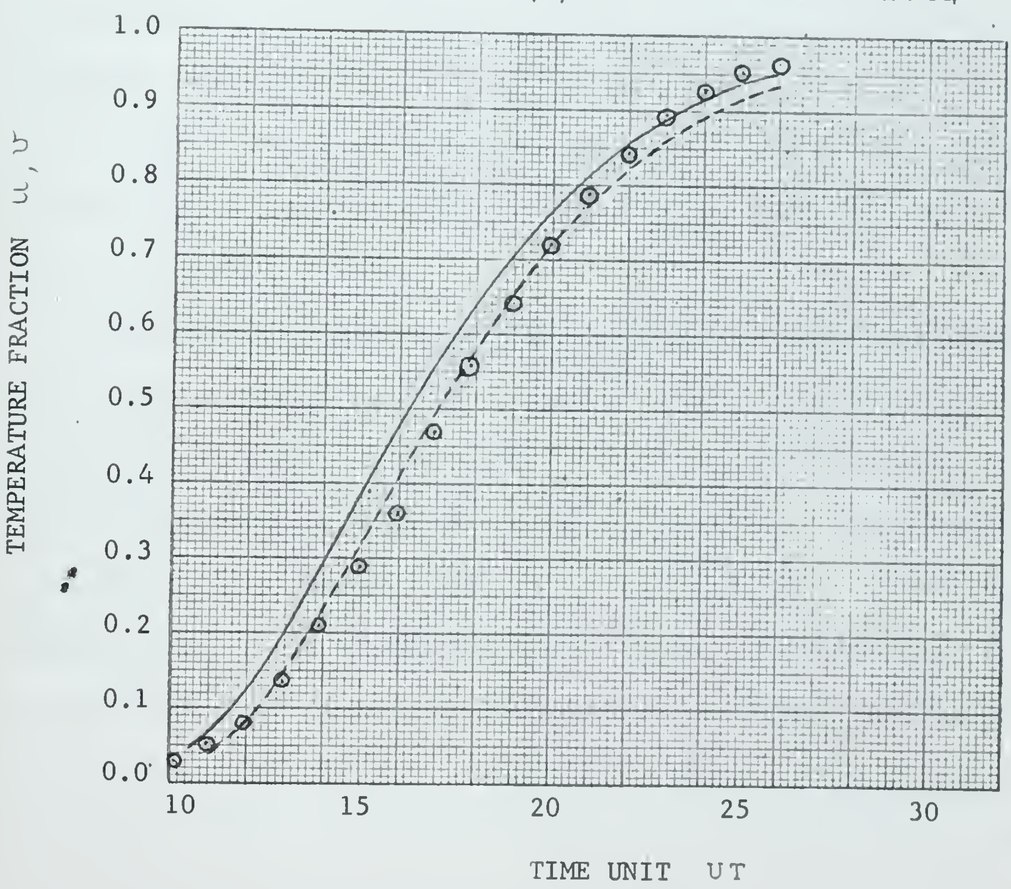


Figure 7. Comparison of numerical temperature profile to PRESTON'S experimental data.

POROUS SYSTEM : MESH OF GLASS-ISO-OCTANE

k_e^o	Btu/hr ft^2 °F/ft	=	0.239
k_s	"	=	0.355
k_f	"	=	0.088
ρ_s	lb/ft^3	=	139.2
ρ_f	"	=	43.0
c_s	Btu/lb°F	=	.1839
c_f	"	=	.5305
ϕ		=	.425
ha	Btu/ft^3 °F, hr	=	277.
X	ft	=	0.958
F	Time unit/hr	=	15.
Vel	ft/hr	=	2.14



——— Fluid temperature

- - - - - Solid temperature

⊙ Experimental data of Preston

6. Discussion of Results

Figure 1 shows temperature profiles for different values of dimensionless parameter λ . Different values of τ are used to prevent the curves from falling on top of each other. These curves agree very well with the results of Green and Perry's calculated by finite difference methods. It is seen that as λ increases, the temperature lag between the two phases decreases. This point can be easily explained by the fact that the dimensionless parameter λ is proportional to the heat transfer coefficient $(ha)^{\frac{1}{2}}$ and to the fluid thermal conductivity, and is inversely proportional to the fluid velocity. Thus, the temperature lag between the two phases decreases for large values of ha or k_f and at low fluid velocities.Thus one can conclude that at very low flow rate, as in the case of oil reservoirs, the approximation $T_f = T_s$ is reasonable; on the contrary, it is not applicable to the case of heat exchangers where the gas flows at high velocity. For very large values of ha, one can assume that the fluid and solid temperature are equal. In this case, the general partial differential equations are reduced to one equation in T.

The effect of the heat transfer coefficient on the temperature lag can be easily studied in Figure 2 where results from numerical solutions to the general differential equations are compared to results from simplified analytical solutions using the equation of Jenkins and Aronofsky and of Schumann. The writer's results also agree with the results of Green and Perry. The graph shows that for values of λ equal or less than .114, the curve for ha, ks and k_f finite approaches the curve obtained by neglecting the thermal conductivities k_s and k_f. For values of λ larger than .342, the numerical solutions become closer to the solutions based on the assumption that only the conduction is the

important heat transfer mechanism. The curves also show that in the intermediate range of λ , both conduction and convection are important and should be considered.

Figure 3 shows the effect of thermal conductivity of the solid phase. For values of $\frac{\alpha_s}{\alpha_f}$ equal or less than .883, both the simplified analytical solutions are acceptable. As the ratio $\frac{\alpha_s}{\alpha_f}$ increases, the assumption of $T_f = T_s$ is increasingly better.

Figure 4 shows the effect of dimensionless parameter λ on the time-temperature history. Decreasing ha or increasing fluid velocity makes the temperature at a given point more responsive. The same results can be obtained by increasing the solid thermal conductivity.

In order to compare numerical solution of the general case to experimental results, the data of Preston was used. The parameters needed for calculation are the thermal conductivities k_s and k_f, the densities ρ_s and ρ_f , the specific heats C_s and C_f, the porosity ϕ , the heat transfer coefficient ha. It should be pointed out that k_s is not a true but pseudo-thermal conductivity characterizing the rate of apparent solid phase conduction and that k_f is the sum of two terms:

$$k_f = k_{fc} + k_{fm}$$

where k_{fc} is the fluid molecular conductivity and k_{fm} is the term characterizing the effect of the eddy-dispersion. This dispersion effect is due to the irregularities of fluid flow in packed beds causing a convective mixing process. The values used for the parameters were furnished by Preston's data [24] , except k_s and k_f. In his experimental work, Preston measured the thermal conductivity k_e° of

the porous medium under no flow condition which he called static thermal conductivity, and the effective or dynamic thermal conductivity of the porous system under flow condition. Using this concept of static thermal conductivity, Green and Perry [7] assumed that this conductivity is the sum of two terms independent of the fluid velocity:

$$k_e^o = k_{fc}\phi + k_s(1-\phi)$$

Thus k_s could be calculated from this relation, knowing the values of k_{fc}, k_e^o and ϕ .

The mixing term of the fluid conductivity could be calculated by two methods:

(1) By assuming that the heat transfer Peclet number N_{pe} is defined as $\frac{V_f d_p \rho_f c_f}{k_{fm}}$ and equal to the mass transfer N_{pe}, one can use the plot of N_{pe} vs N_{re} available in mass transfer experimental work in porous media [6] to get k_{fm}.

(2) By using the correlating equation derived by Green and Perry and Babcock from experimental data [8] :

$$\frac{k_{fm}}{k_{fc}} = 0.115\left(\frac{V_f d_p}{D}\right)$$

where D is the molecular diffusivity $\frac{k_{fc}}{\rho_f c_f}$ for fluid phase heat transfer.

Since the reference [6] was not available in time, the correlating equation in method (2) was used to calculate k_{fm}. The result is not reliable, for Green and Perry state that this equation should be applied only to the values of $V_f d_p$ greater than 0.03.

Temperature history was plotted for two systems of packed bed. Fig. 6

shows that numerical results agree with experimental data. In Figure 7, the solid temperature curves approaches the experimental data while the fluid temperature curve is not too close. This might be due to the fact that the value of $V_f d_p$ in this case is beyond the range for which the application of the correlating equation is valid.

7. Conclusions

The following conclusions may be drawn from the results discussed in the preceding section:

(1) Approximating the fluid and solid temperatures by the same value is reasonable only in the range of very low fluid velocities. This confirms the conclusion of Preston [24] who stated that at velocities less than 0.05 ft/hr, the effective thermal conductivity under flow conditions is equal to the thermal conductivity of the system measured without fluid flowing.

(2) The approximation of $T_f = T_s$ is still applicable to the porous systems which have large heat transfer coefficients.

(3) The fluid velocity considerably affects the temperature profiles.

(4) At high rate of fluid flow, the heat transfer coefficient plays a predominant role (it increases with velocity).

(5) Salzer's method of numerical inversion of Laplace transforms may be very helpful for the solution of a system of partial differential equations with constant coefficients. It was shown that this method is much faster than other numerical inversion methods using Legendre, Laguerre and Chebyscheff polynomials; it has also the advantage over the finite difference methods which require small increments in space and in time for acceptable accuracy. The only problem encountered in this method was that the convergence of the integration could not be obtained as the order of polynomials was increased. But the results were considered satisfactory since values of the inverse transform agree to three significant figures for all orders from 4 to 16. Numerical inversion methods of Laplace transforms are still in development and promise to be the main alternative to the finite difference method.

8. Recommendations for Future Studies

(1) Green and Perry suggested that an effective thermal conductivity of a porous system can be obtained by selecting a value of k_e which will give the best agreement between the numerical temperature profile and the analytical solution to the equation considering $T_f = T_s$. This may be done by comparing the maximum slope of the two curves. A prediction of the behavior of the slope might be helpful; it could be made by examining the formula giving the slope of equation (5) based on the assumption $T_f = T_s$. This formula derived by Preston [24] is:

$$\frac{dv}{d\theta} = \frac{x}{2\theta\sqrt{\pi K\theta}}\, e^{-\frac{1}{4K}\left(\frac{x}{\theta} - \frac{V_f C_1 \sqrt{\theta}}{C_3}\right)^2}$$

where
$$K = \frac{k_e}{\rho_f c_f \phi + \rho_s c_s (1-\phi)}$$

and k_e is the effective thermal conductivity of the system. It is shown from the slope formula that the slope decreases as k_e increases. Results from this recommended investigation may be compared to experimental data of Preston and then may verify the validity of the suggestion of Green and Perry.

(2) Skobl$\widehat{\text{ia}}$'s new method of numerical inversion of Laplace transforms [30] may be used to solve this problem and make comparisons of the two methods. It is expected that Skobl$\widehat{\text{ia}}$'s method will give more accurate results.

(3) A subroutine may be written for the case of heat regenerators and packed beds of finite length, using the mathematical derivations in

Appendix III. Thus the effect of the heat transfer parameters on temperature profiles, values of NTU and effectiveness for heat regenerators may be investigated. Results may be compared to the results obtained by Creswick [5], Moreland [20], Bahnke [2] and Nahavandi and Weinstein [21]. However, note the remark at the end of Appendix III.

BIBLIOGRAPHY

1. Anzelius, A., "Über Erwärmung vermittels durchströmender Medien", Z. Angen. Math. Mech. 6, (1926) pp. 291-294.

2. Bahnke, G. D., "Effect of Longitudinal Heat Transfer Conductivity on Rotary Regenerators", Master Thesis, USNPGS, (1962).

3. Carslaw and Jaeger, Operational Methods in Applied Mathematics, Dover Publications, Inc., New York (1963).

4. Cost, T. L., "Approximate Laplace Transform Inversions in Viscoelastic Stress Analysis", AIAA Journal, Vol. 2, 12, (1964), pp. 2157-2166.

5. Creswick, F. A., "A Digital Computer Solution of the Equations for Transient Heating of a Porous Solid Including the Effects of Longitudinal Conduction", Industrial Mathematics, (1957), pp. 61-69.

6. Ebach, E. A. and R. R. White, A.I. Ch.E. Journal, 4, 161 (1958).

7. Green, D. W. and R. H. Perry, "Heat Transfer with a Flowing Fluid through Porous Media", Chemical Eng. Progr. Symp. Series, No. 32, 57, (1961) pp 61-68.

8. Green, D. W., Perry, R. H., Babcock, R. E., "Longitudinal Dispersion of Thermal Energy through Porous Media with a Flowing Fluid", A.I.Ch.E. Journal, Vol. 10, No. 5, Sept (1964), pp. 645-651.

9. Hadidi, T. A. R., Ph.D. Thesis, The Pennsylvania State University (1955).

10. Hausen, H., Näherungsverfahren zur Berechnung des Wärmeaustausches in Regeneratoren, Z. Angen. Math. Mech. 2, (1931), pp. 105-114.

11. Hurwitz, H., Jr., and P. F. Sweifel, "Numerical Quadrature of Fourier Transform Integrals", Math. Tables Aids Comp., X, (1956), pp. 140-149.

12. Iliffe, C. E., "Thermal Analysis of the Counterflow Regenerative Heat Exchanger", J. Inst. Mech. Engrs., 159, (1948), pp. 363-372.

13. Jakob, M., Heat Transfer, Vol. II, John Wiley and Sons, Inc., (1963)

14. Jenkins, R. and J. S. Aronofsky, "Analysis of Heat Transfer Processes in Porous Media. New Concepts in Reservoir Heat Engineering. Mineral Industries Experiment Station Bulletin 64, p. 69 (1954). The Pennsylvania State University, State College, Pa.

15. Jenkins, R. and J. S. Aronofsky, Producers Monthly, 19, 37, (1955).

16. Krall, H. L. and O. Frink, "A New Class of Orthogonal Polynomials: The Bessel Polynomials", Trans. Amer. Math. Soc., 65,(1949), pp.100-115.

17. Lambertson, T. J., Performance Factors of a Periodic-Flow Heat Exchanger, Trans. A.S.M.E., 159,(1958), pp 586-592.

18. Lanczos, C., Applied Analysis, Prentice Hall, Inc., (1956).

19. Locke, G. L., "Heat Transfer and Flow-Friction Characteristics of Porous Solids", Department of Mechanical Engineering, Stanford University, TR No. 10, June (1950).

20. Moreland, F. E., "Solution of the Single Blow Problem with Longitudinal Conductivity by Numerical Inversion of Laplace Transforms", Master Thesis, USNPGS (1964).

21. Nahavandi, A. N. and A. S. Weinstein, "A Solution to the Periodic Flow Regenerative Heat Exchanger Problem", Applied Scientific Research, Section A, 10, (1961) pp 335-348.

22. Nusselt, W., Der Beharrungszustand im Winderhitzen, Z. Ver. dtsch. Ing. 72, (1928), pp.1052-1054.

23. Papoulis, A., "A New Method of Inversion of the Laplace Transform", Quart. Appl. Math. 14, (1957), pp 405-414.

24. Preston, F. W., Ph.D Thesis, Pennsylvania State University (1957).

25. Salzer, H. E., "Orthogonal Polynomials Arising in the Numerical Evaluation of Inverse Laplace Transforms", Math. Tables Aids Comput. 9,(1955), pp 164-177.

26. Salzer, H. E., "Tables for the Numerical Calculation of Inverse Laplace Transforms", J. Math. and Physics, 37,(1958), pp 89-108.

27. Salzer, H. E., "Additional Formulas and Tables for Orthogonal Polynomials Originating from Inversion Integrals", J. Math. Phys. Vol. 40,(1961), pp 72-86.

28. Schmittroth, L. A., "Numerical Inversion of Laplace Transforms", Communications of the ACM, pp 171-172.

29. Schumann, T.E.W., "Heat Transfer: A Liquid Flowing Through a Porous Prism", J. Franklin Institute, 208,(1929), pp 405-416.

30. Skobli͡a, N., "Tables for the Numerical Inversion of Laplace Transforms", Academy of Sciences of USSR, Moscow,(1964).

31. Willmott, A. J., "Digital Computer Simulation of a Thermal Regenerator", Intern. J. Heat Mass Transf. Vol. 7,(1964), pp.1291-1302.

APPENDIX I

PARTICULAR CASE WHERE LONGITUDINAL CONDUCTION IN BOTH THE FLUID AND SOLID ARE NEGLECTED.

The terms describing longitudinal conduction are neglected and the differential equations (11) and (12) in the general case are reduced to:

$$\frac{\partial u}{\partial \tau} = (\lambda\beta\gamma)(v - u) \tag{27}$$

$$\frac{\partial v}{\partial \tau} = -\frac{\partial v}{\partial y} - \lambda(v - u) \tag{28}$$

Let $\lambda\beta\gamma = b$

the Laplace transform of these 2 equations are:

$$s\bar{u} = b(\bar{v} - \bar{u}) \tag{29}$$

$$s\bar{v} = -\frac{\partial \bar{v}}{\partial y} - \lambda(\bar{v} - \bar{u}) \tag{30}$$

From equation (29):

$$\bar{u} = \frac{b\bar{v}}{s + b}$$

Replacing $\bar{u}$ in (30) by its value, we get:

$$s\bar{v} = -\frac{d\bar{v}}{dy} - \left(\lambda\bar{v} - \frac{\lambda b\bar{v}}{s + b}\right)$$

or

$$\frac{d\bar{v}}{dy} + \left(s + \lambda - \frac{\lambda b}{s + b}\right)\bar{v} = 0 \tag{31}$$

Solving this ordinary differential equation yields:

$$\bar{v} = C \exp\left[-(s+\lambda)y + \frac{\lambda b}{s+b} y\right]$$

With the boundary condition $\bar{v}(o,s) = \frac{1}{s}$, we get:

$$\bar{v} = \frac{1}{s} \exp\left[-(s+\lambda)y + \frac{\lambda b}{s+b} y\right] \tag{32}$$

and

$$\bar{u} = \frac{b}{s(s+b)} \exp\left[-(s+\lambda)y + \frac{\lambda b}{s+b} y\right] \tag{33}$$

From a table of Laplace transforms, we get the formula:

$$\frac{1}{s^{\mu}} e^{\frac{\alpha}{s}} = \mathcal{L}\left\{\left(\frac{\tau}{\alpha}\right)^{\left(\frac{\mu-1}{2}\right)} I_{\mu-1}\left[2\sqrt{\alpha\tau}\right]\right\}$$

For $\mu = 1$

$$\frac{1}{s} e^{\frac{\alpha}{s}} = \mathcal{L}\left\{ I_o\left[2\sqrt{\alpha\tau}\right]\right\}$$

A theorem of Laplace transforms states that if a is any constant and

$$\mathcal{L}\left\{f(\tau)\right\} = F(s)$$

then $$\mathcal{L}\left\{e^{-a\tau} f(\tau)\right\} = F(s+a)$$

Hence $$\frac{1}{s+b} e^{\frac{\alpha}{s+b}} = \mathcal{L}\left\{e^{-b\tau} I_o\left[2\sqrt{\alpha\tau}\right]\right\} \tag{34}$$

From a theorem on the Laplace transform of an integral, we have

$$\mathcal{L}\left\{\int_0^\tau f(\xi)\,d\xi\right\} = \frac{1}{s}\mathcal{L}\left\{f(\tau)\right\}$$

It then follows that:

$$\frac{1}{s(s+b)}\,e^{\frac{\alpha}{s+b}} = \mathcal{L}\left\{\int_0^\tau e^{-b\xi}\,I_0\left[2\sqrt{\alpha\xi}\right]d\xi\right\} \tag{35}$$

But

$$\frac{1}{s(s+b)}\,e^{\frac{\alpha}{s+b}} = \frac{1}{b}\left(\frac{1}{s} - \frac{1}{s+b}\right)e^{\frac{\alpha}{s+b}} \tag{36}$$

Hence from (34), (35) and (36), we get:

$$\frac{1}{s}e^{\frac{\alpha}{s+b}} = \mathcal{L}\left\{e^{-b\tau}\,I_0\left[2\sqrt{\alpha\tau}\right] + b\int_0^\tau e^{-b\xi}\,I_0\left[2\sqrt{\alpha\xi}\right]d\xi\right\} \tag{37}$$

From properties of Bessel functions, we find:

$$I_0'[z] = I_1[z]$$

then by integrating by parts, the integral of equation (37) becomes:

$$\frac{1}{b}\int_0^\tau e^{-b\xi}\,I_0\left[2\sqrt{\alpha\xi}\right]d\xi = -\,e^{-b\tau}\,I_0\left[2\sqrt{\alpha\tau}\right] + 1 +$$

$$+\sqrt{\alpha}\int_0^\tau e^{-b\xi}\,I_1\left[2\sqrt{\alpha\xi}\right]\xi^{-\frac{1}{2}}\,d\xi \tag{38}$$

From equations (37) and (38):

$$\frac{1}{s} e^{\frac{\alpha}{s+b}} = \mathcal{L}\left\{ 1 + \sqrt{\alpha} \int_0^{\tau} e^{-b\xi} I_1\left[2\sqrt{\alpha\xi}\right] \xi^{-\frac{1}{2}} d\xi \right\} \tag{39}$$

The translation theorem of Laplace transforms states that if

$$f(\tau) = H(\tau - a)\phi(\tau - a)$$

where $H(\tau - a)$ is Heaviside's unit step function defined by

$$H(\tau - a) = 0 \quad \text{for} \quad \tau < a$$

$$H(\tau - a) = 1 \quad \text{for} \quad \tau > a$$

then

$$\mathcal{L}\left\{f(\tau)\right\} = e^{-as} \mathcal{L}\left\{\phi(\tau)\right\} \tag{40}$$

From equation (32), the transform of v can be written as:

$$\bar{v} = e^{-\lambda y} \left[e^{-ys} \cdot \frac{1}{s} e^{\frac{\lambda b}{s+b} y} \right] \tag{41}$$

From equations (39), (40) and (41), it is found that:

$$\phi(\tau) = \mathcal{L}^{-1}\left\{ \frac{1}{s} e^{\frac{\lambda b}{s+b} y} \right\} = 1 + \sqrt{\lambda b y} \int_0^{\tau} e^{-b\xi} I_1\left[2\sqrt{\lambda b y \xi}\right] \xi^{-\frac{1}{2}} d\xi$$

and

$$\phi(\tau - y) = 1 + \sqrt{\lambda b y} \int_0^{\tau - y} e^{-b\xi} I_1\left[2\sqrt{\lambda b y \xi}\right] \xi^{-\frac{1}{2}} d\xi$$

then for $\tau > y$, we get:

$$v = e^{-\lambda y} \left\{ 1 + \sqrt{\lambda b y} \int_0^{\tau - y} e^{-b\xi} I_1\left[2\sqrt{\lambda b y \xi}\right] \xi^{-\frac{1}{2}} d\xi \right\} \tag{42}$$

Eq (33) can be written as:

$$\bar{u} = e^{-\lambda y} \left[e^{-ys} \cdot \frac{b}{s(s+b)} e^{\frac{\lambda b}{s+b} y} \right] \tag{43}$$

Using equations (35), (40) and (43), we can write:

$$u = b e^{-\lambda y} \int_0^{\tau - y} e^{-b\xi} I_0\left[2\sqrt{\lambda b y \xi}\right] d\xi \tag{44}$$

The equations (42) and (44) were evaluated by the program SCHUMANN. Numerical results were checked against solutions from numerical inversion of equations (32) and (33). The agreement was to three significant figures.

A more simplified solution can be obtained by neglecting the term $\frac{\partial v}{\partial \tau}$ in equation (28) which describes the energy stored in the fluid. Thus the equations (27) and (28) become:

$$\frac{\partial u}{\partial \tau} = b(v - u) \tag{45}$$

$$\frac{\partial v}{\partial y} = -\lambda(v - u) \tag{46}$$

transforming the above equations yields:

$$s\bar{u} = b(\bar{v} - \bar{u}) \tag{47}$$

$$\frac{\partial \bar{v}}{\partial y} = -\lambda(\bar{v} - \bar{u}) \tag{48}$$

From (47):

$$\bar{u} = \frac{b\bar{v}}{s + b} \tag{49}$$

Substituting this value of $\bar{u}$ in (48), we get the ordinary differential equation in $\bar{v}$:

$$\frac{d\bar{v}}{dy} + \left(\lambda - \frac{\lambda b}{s+b}\right)\bar{v} = 0$$

The solution is:

$$\bar{v} = Ce^{-\left(\lambda - \frac{\lambda b}{s+b}\right)y}$$

With $\bar{\upsilon}(o,s) = \frac{1}{s}$, we find:

$$\bar{\upsilon} = \frac{1}{s} e^{-\left(\lambda - \frac{\lambda b}{s+b}\right)y} \tag{50}$$

Using equation (39), the inverse transform of (50) can be written as:

$$\upsilon = e^{-\lambda y}\left\{1 + \sqrt{\lambda b y}\int_0^{\tau} e^{-b\xi}\, I_1\left[2\sqrt{\lambda b y \xi}\right]\xi^{-\frac{1}{2}}\, d\xi\right\} \tag{51}$$

From eq. (49) and (50), it follows that:

$$\bar{u} = e^{-\lambda y}\left[\frac{b}{s(s+b)}\, e^{\frac{\lambda b}{s+b} y}\right] \tag{52}$$

Using equation (35) yields the inverse transform of (52):

$$u = e^{-\lambda y}\int_0^{\tau} e^{-b\xi}\, I_o\left[2\sqrt{\lambda b y \xi}\right] d\xi \tag{53}$$

The solution to this simplified case has been derived by Moreland [20] in the form of a series. His solution can not be evaluated at y = 0. The equation (51) was programmed and checked against Moreland's solution. The agreement is good up to 8 significant figures.

APPENDIX II

PARTICULAR CASE WHERE THE FLUID-SOLID BOUNDARY RESISTANCE IS NEGLIGIBLE, I.E., ha = infinite.

In this case, ha infinite is equivalent to assuming that the fluid and solid temperatures are equal at any time throughout the bed.

Combining the equations (11) and (12) and letting $v = u$ yields the following equation:

$$\left(1+\frac{1}{\beta\gamma}\right)\frac{\partial v}{\partial \tau} = -\frac{\partial v}{\partial y} + \left(1+\frac{1}{\gamma}\right)\lambda\frac{\partial^2 v}{\partial y^2} \tag{54}$$

Let $c = 1 + \frac{1}{\beta\gamma}$

$a = \left(1+\frac{1}{\gamma}\right)\lambda$

The transform of equation (54) is:

$$cs\bar{v} = -\frac{d\bar{v}}{dy} + a\frac{d^2\bar{v}}{dy^2}$$

or

$$a\frac{d^2\bar{v}}{dy^2} - \frac{d\bar{v}}{dy} - cs\bar{v} = 0$$

The characteristic equation is then:

$$ar^2 - r - cs = 0$$

This is a quadratic equation with complex coefficients. The roots may be written as:

$$r = \frac{1}{2a} \pm \sqrt{\frac{1}{4a^2} + \frac{c}{a} s}$$

$$r = \frac{1}{2a} \pm \sqrt{\frac{1}{p}(q + s)} \tag{55}$$

where

$$p = \frac{a}{c}$$

$$q = \frac{1}{4ac}$$

Hence:

$$\bar{v} = C_1 e^{r_1 y} + C_2 e^{r_2 y} \tag{56}$$

Applying the boundary condition at y = ∞ , we have:

$$\bar{v}(\infty, s) = 0$$

Thus only the roots with negative real parts may be used.

Suppose that r_1 has a negative real part,

then

$$\bar{v}(y, s) = C_1 e^{r_1 y}$$

Since

$$\bar{v}(0,s) = \frac{1}{s} e^{r_1 y}$$

then

$$\bar{v}(y,s) = e^{\frac{1}{2a}y} \left[\frac{1}{s} e^{-y\sqrt{\frac{1}{p}(s+q)}} \right]$$

From table of Laplace transform we have:

$$\mathcal{L}^{-1}\left\{ \frac{e^{-x\sqrt{\frac{s}{p}}}}{s-q} \right\} = \frac{1}{2} e^{qt} \left\{ e^{-x\sqrt{\frac{q}{p}}} \operatorname{erfc}\left[\frac{x}{2\sqrt{pt}} - \sqrt{qt} \right] + \right.$$

$$\left. + \; e^{x\sqrt{\frac{q}{p}}} \operatorname{erfc}\left[\frac{x}{2\sqrt{pt}} + \sqrt{qt} \right] \right\}$$

Since

$$\mathcal{L}^{-1}\left\{ f(s+q) \right\} = e^{-qt} f(t)$$

then:

$$\mathcal{L}^{-1}\left\{ \frac{1}{s} e^{-x\sqrt{\frac{1}{p}(s+q)}} \right\} = \frac{1}{2} \left\{ e^{-x\sqrt{\frac{q}{p}}} \operatorname{erfc}\left[\frac{x}{2\sqrt{pt}} - \sqrt{qt} \right] + \right.$$

$$\left. + \; e^{x\sqrt{\frac{q}{p}}} \operatorname{erfc}\left[\frac{x}{2\sqrt{pt}} + \sqrt{qt} \right] \right\}$$

and

$$v(y,\tau) = \frac{1}{2}\left\{ \operatorname{erfc}\left[\frac{y}{2\sqrt{p\tau}} - \sqrt{q\tau}\right] + e^{2y\sqrt{\frac{q}{p}}} \operatorname{erfc}\left[\frac{y}{2\sqrt{p\tau}} + \sqrt{q\tau}\right] \right\} \tag{57}$$

The equation (57) was programmed by using the subroutine ERFN. After some testing runs, it was found that for large values of y, the second term in equation (57) did not give accurate results for the reason that the exponential term becomes very large and the complementary error function becomes very small. Thus the product of these two functions certainly gives large error. The error can be minimized by approximating the complementary error function as a series. A well known asymptotic series for erfc x is:

$$\operatorname{erfc} x = \frac{e^{-x^2}}{\sqrt{\pi}}\left(\frac{1}{x} - \frac{1}{2x^3} - \frac{1.3}{2^2 x^5} - \frac{1.3.5}{2^3 x^7} + \text{------} \right)$$

Thus the equation (57) may be replaced by:

$$v(y,\tau) = \frac{1}{2}\left\{ \operatorname{erfc}\left[\frac{y}{2\sqrt{p\tau}} - \sqrt{q\tau}\right] + \frac{e^{2y\sqrt{\frac{q}{p}} - x^2}}{\sqrt{\pi}} \sum_{n=1}^{\infty} \frac{(2n-3)!}{2^{n-1} x^{2n-1}} \right\} \tag{58}$$

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

where

$$x = \frac{y}{2\sqrt{p\tau}} + \sqrt{q\tau}$$

$$p = \frac{\beta\lambda(1+\gamma)}{(1+\beta\gamma)}$$

$$q = \frac{\beta\gamma^2}{4\lambda(1+\gamma)(\beta\gamma+1)}$$

The equation (58) was evaluated by the program JENKINS. Solutions from this program were compared with results from the programs TEMFLU1 and SCHUMANN. Discussions of these results are presented in Section 6.

APPENDIX III

GENERAL CASE APPLIED TO A MODEL OF FINITE LENGTH.

The following mathematical derivations are applied to heat regenerators and packed beds of finite length.

The energy balance is as follows:

a. For the fluid phase:

Heat stored in an element of fluid : $\rho_f A_f c_f \frac{\partial T_f}{\partial \theta}$

Convection by moving fluid : $\dot{\omega}_f c_f \frac{\partial T_f}{\partial x}$

Conduction in the fluid : $k_f A_f \frac{\partial^2 T_f}{\partial x^2}$

Heat transferred to the fluid element by convection : $\frac{hA}{L}(T_f - T_s)$

then:

$$\rho_f A_f c_f \frac{\partial T_f}{\partial \theta} = - \dot{\omega}_f c_f \frac{\partial T_f}{\partial x} + k_f A_f \frac{\partial^2 T_f}{\partial x^2} - \frac{hA}{L}(T_f - T_s) \tag{59}$$

(b) For the solid phase:

Heat gained by an element of solid : $\rho_s A_s c_s \frac{\partial T_s}{\partial \theta}$

Heat transferred to the solid element by convection : $\frac{hA}{L}(T_f - T_s)$

Heat transferred by conduction from the solid element : $k_s A_s \frac{\partial^2 T_s}{\partial x^2}$

then:

$$\rho_s A_s c_s \frac{\partial T_s}{\partial \theta} = k_s A_s \frac{\partial^2 T_s}{\partial x^2} + \frac{hA}{L}(T_f - T_s) \tag{60}$$

Multiplying (59) and (60) by $\frac{L}{ha}$:

$$(\rho_f A_f c_f)\left(\frac{L}{hA}\right)\frac{\partial T_f}{\partial \theta} = -\dot{\omega}_f c_f \left(\frac{L}{hA}\right)\frac{\partial T_f}{\partial x} + \left(k_f A_f \frac{L}{hA}\right)\frac{\partial^2 T_f}{\partial x^2} - (T_f - T_s) \tag{61}$$

$$(\rho_s A_s c_s)\left(\frac{L}{hA}\right)\frac{\partial T_s}{\partial \theta} = k_s A_s \frac{L}{hA}\frac{\partial^2 T_s}{\partial x^2} + (T_f - T_s) \tag{62}$$

Let us define the following parameters:

$X = \frac{x}{L}$ dimensionless length parameter

$t = \frac{hA\theta}{W_s c_s}$ dimensionless time parameter

$\lambda' = \frac{k_s A_s}{\dot{\omega}_f c_f L}$ dimensionless conduction parameter

$NTU = \frac{hA}{\dot{\omega}_f c_f}$ dimensionless heat transfer unit

Substituting these parameters in (61) and (62) yields:

$$(\rho_f A_f c_f)\left(\frac{L}{W_s c_s}\right)\frac{\partial T_f}{\partial t} = -\left(\frac{\dot{\omega}_f c_f}{hA}\right)\frac{\partial T_f}{\partial X} + \left(\frac{k_f A_f}{hAL}\right)\frac{\partial^2 T_f}{\partial X^2} - (T_f - T_s) \tag{63}$$

and $$\frac{\partial T_s}{\partial t} = \left(\frac{\lambda'}{NTU}\right)\frac{\partial^2 T_s}{\partial x^2} + (T_f - T_s) \tag{64}$$

Multiplying (63) by $\dfrac{W_s c_s}{\rho_f A_f c_f L}$, we get:

$$\frac{\partial T_f}{\partial t} = -\left(\frac{W_s c_s}{\rho_f A_f c_f L}\right)\left(\frac{\dot{w}_f c_f}{hA}\right)\frac{\partial T_f}{\partial x} + \left(\frac{W_s c_s}{L^2 hA}\right)\left(\frac{k_f}{\rho_f c_f}\right)\frac{\partial^2 T_f}{\partial x^2} - \frac{W_s c_s}{\rho_f A_f c_f L}(T_f - T_s) \tag{65}$$

Let us define:

$$\alpha = \frac{k}{\rho c} \quad = \text{thermal diffusivity}$$

$$\beta' = \frac{\alpha_f}{\alpha_s}$$

$$\psi = \frac{\rho_s c_s A_s}{\rho_f c_f A_f} \quad = \text{ratio of heat capacities per unit length}$$

Substituting these parameters in equation (65) yields:

$$\frac{\partial T_f}{\partial t} = -\left(\frac{\psi}{NTU}\right)\frac{\partial T_f}{\partial x} + \left(\frac{\beta'\lambda'}{NTU}\right)\frac{\partial^2 T_f}{\partial x^2} - \psi(T_f - T_s) \tag{66}$$

If we define:

$$u = \frac{T_s}{T_i} \quad \text{and} \quad v = \frac{T_f}{T_i}$$

where T_i is the step input injected fluid temperature, the equations (64) and (66) become:

$$\frac{\partial u}{\partial t} = a \frac{\partial^2 u}{\partial X^2} + (v - u) \tag{67}$$

$$\frac{\partial v}{\partial t} = a\beta' \frac{\partial^2 v}{\partial X^2} - b\frac{\partial v}{\partial X} - \psi(v - u) \tag{68}$$

where $$a = \frac{\lambda'}{NTU}$$

$$b = \frac{\psi}{NTU}$$

The initial conditions and boundary conditions are assumed as follows:

(a) Initial conditions:

The initial fluid and matrix temperature are uniform and equal. The base scale temperature can be chosen so that these temperatures can be taken as zero for convenience:

$$u(X,0) = v(X,0) = 0$$

(b) Boundary conditions:

(1) At X = 0 and t = 0^+, the injected fluid temperature is suddenly changed to a different higher value and held constant thereafter:

$$v(0,t) = 1$$

(2) At X = 0 and t = 0^+, the solid temperature instantaneously rises to the value of the step input temperature of the fluid:

$$u(0, t) = 1$$

(3) The matrix is insulated at X = 0:

$$\frac{\partial T_s}{\partial x}(0,t) = 0$$

(4) The matrix is also insulated at X = 1:

$$\frac{\partial T_s}{\partial x}(1,t) = 0$$

From equation (68) we have:

$$u = \frac{1}{\psi}\left[\frac{\partial v}{\partial t} - a\beta'\frac{\partial^2 v}{\partial x^2} + b\frac{\partial v}{\partial x} + \psi v\right] \qquad (69)$$

$$\frac{\partial u}{\partial t} = \frac{1}{\psi}\left[\frac{\partial^2 v}{\partial t^2} - a\beta'\frac{\partial^3 v}{\partial x^2 \partial t} + b\frac{\partial^2 v}{\partial x \partial t} + \psi\frac{\partial v}{\partial t}\right] \qquad (70)$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{\psi}\left[\frac{\partial^3 v}{\partial x^2 \partial t} - a\beta'\frac{\partial^4 v}{\partial x^4} + b\frac{\partial^3 v}{\partial x^3} + \psi\frac{\partial^2 v}{\partial x^2}\right] \qquad (71)$$

Substituting equations (69), (70) and (71) in (67) and rearranging the terms, we get:

$$\left(\frac{a^2\beta'}{\psi}\right)\frac{\partial^4 v}{\partial x^4} - \left(\frac{ab}{\psi}\right)\frac{\partial^3 v}{\partial x^3} - \frac{a}{\psi}\left(1+\beta'\right)\frac{\partial^3 v}{\partial x^2 \partial t} - a\left(1+\frac{\beta'}{\psi}\right)\frac{\partial^2 v}{\partial x^2} +$$

$$+\left(\frac{b}{\psi}\right)\frac{\partial v}{\partial x} + \left(\frac{b}{\psi}\right)\frac{\partial^2 v}{\partial x \partial t} + \left(1+\frac{1}{\psi}\right)\frac{\partial v}{\partial t} + \frac{1}{\psi}\frac{\partial^2 v}{\partial t^2} = 0 \qquad (72)$$

The transform of equation (72) is:

$$\left(\frac{a^2\beta'}{\psi}\right)\frac{d^4\bar{v}}{dx^4} - \left(\frac{ab}{\psi}\right)\frac{d^3\bar{v}}{dx^3} - \frac{a}{\psi}\left[(1+\beta')s + (\beta'+\psi)\right]\frac{d^2\bar{v}}{dx^2} +$$

$$+ \frac{b}{\psi}(1+s)\frac{d\bar{v}}{dx} + \left(\frac{s^2}{\psi} + \frac{s}{\psi} + s\right)\bar{v} = 0 \qquad (73)$$

the corresponding auxiliary equation is then:

$$A_4 r^4 + A_3 r^3 + A_2 r^2 + A_1 r + A_0 = 0 \qquad (74)$$

Where $A_0, A_1, \ldots, A_4$ are the complex coefficients of equation (73). The general solution in the Laplace S plane for the fluid temperature is:

$$\bar{v} = C_1(s)e^{r_1 x} + C_2(s)e^{r_2 x} + C_3(s)e^{r_3 x} + C_4(s)e^{r_4 x} \qquad (75)$$

where r_1, r_2, r_3, r_4 are the complex roots of equation (74). The boundary conditions are transformed and then used to determine the coefficients C_n:

BC.1 $\qquad \bar{v}(o,s) = \frac{1}{s}$

BC.2 $\qquad \bar{u}(o,s) = \frac{1}{s}$

BC.3 $\quad \frac{\partial \bar{u}}{\partial x}(o,s) = 0$

BC.4 $\quad \frac{\partial \bar{u}}{\partial x}(1,s) = 0$

Applying BC.1 to equation (75) gives:

$$\bar{v}(0,s) = \sum_{n=1}^{4} C_n(s) = \frac{1}{s} \tag{76}$$

Taking the derivatives of $\bar{v}$ (x,s) with respect to x and evaluating them at x = 0 yields:

$$\frac{\partial \bar{v}}{\partial x}(0,s) = \sum_{n=1}^{4} r_n C_n(s) \tag{77}$$

$$\frac{\partial^2 \bar{v}}{\partial x^2}(0,s) = \sum_{n=1}^{4} r_n^2 C_n(s) \tag{78}$$

$$\frac{\partial^3 \bar{v}}{\partial x^3}(0,s) = \sum_{n=1}^{4} r_n^3 C_n(s) \tag{79}$$

From equation (69) we have:

$$\frac{\partial \bar{u}}{\partial x} = \frac{1}{\psi}\left[s\frac{\partial \bar{v}}{\partial x} - a\beta'\frac{\partial^3 v}{\partial x^3} + b\frac{\partial^2 v}{\partial x^2} + \psi\frac{\partial v}{\partial x}\right] \tag{80}$$

Applying the BC (3) to equation (80) and using the equations (77), (78) and (79) give:

$$-a\beta'\sum_{n=1}^{4} r_n^3 C_n(s) + b\sum_{n=1}^{4} r_n^2 C_n(s) + (\psi + s)\sum_{n=1}^{4} r_n C_n(s) = 0 \tag{81}$$

Applying the BC 4 resulting in:

$$-a\beta'\sum_{n=1}^{4} r_n^3 C_n e^{r_n} + b\sum_{n=1}^{4} r_n^2 C_n e^{r_n} + (\psi + s)\sum_{n=1}^{4} r_n C_n e^{r_n} = 0 \quad (82)$$

Let us define the quantity:

$$R_n \equiv \left[-(a\beta')r_n^3 + br_n^2 + (\psi + s)\, r_n\right] \qquad n = 1,2,3,4$$

then the equations (81) and (82) can be written as follows:

$$\sum_{n=1}^{4} R_n C_n = 0 \quad (81a)$$

and

$$\sum_{n=1}^{4} R_n e^{r_n} C_n = 0 \quad (82a)$$

Finally, applying the BC (2) to the transform of u yields:

$$\bar{u}(o,s) = \frac{1}{\psi}\left[-a\beta' \frac{\partial^2 \bar{v}}{\partial x^2}(o,s) + b\frac{\partial \bar{v}}{\partial x} + (s+\psi)\bar{v}(o,s)\right] = \frac{1}{s} \quad (83)$$

or

$$-a\beta'\sum_{n=1}^{4} r_n^2 C_n + b\sum_{n=1}^{4} r_n C_n = -1$$

or

$$\sum_{n=1}^{4} Z_n C_n = -1 \quad (83a)$$

where $Z_n = \left[-a\beta' r_n^2 + br_n\right]$

The equations (76), (81a), (82a) and (83a) are then used to solve for the coefficients $C_n(s)$ of equation (75).

We have the following matrix equation:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ R_1 & R_2 & R_3 & R_4 \\ R_1e^{r1} & R_2e^{r2} & R_3e^{r3} & R_4e^{r4} \\ Z_1 & Z_2 & Z_3 & Z_4 \end{bmatrix} \begin{bmatrix} C_1(s) \\ C_2(s) \\ C_3(s) \\ C_4(s) \end{bmatrix} = \begin{bmatrix} \frac{1}{s} \\ 0 \\ 0 \\ -1 \end{bmatrix} \tag{84}$$

the determinant of the matrix (84) can be written as:

$$\begin{aligned} \Delta = {} & R_2R_3(Z_4-Z_1)(e^{r3}-e^{r2}) + R_2R_4(Z_3 - Z_1)(e^{r2} - e^{r4}) \\ & + R_3R_4(Z_2-Z_1)(e^{r4}-e^{r3}) + R_1R_3(Z_4 - Z_2)(e^{r1} - e^{r3}) \\ & + R_1R_4(Z_2-Z_3)(e^{r1}-e^{r4}) + R_1R_2(Z_4 - Z_3)(e^{r2} - e^{r1}) \end{aligned}$$

the coefficients $C_n(s)$ are then:

$$C_1(s) = \frac{1}{\Delta} \cdot \begin{bmatrix} \frac{1}{s} & 1 & 1 & 1 \\ 0 & R_2 & R_3 & R_4 \\ 0 & R_2e^{r2} & R_3e^{r3} & R_4e^{r4} \\ -1 & Z_2 & Z_3 & Z_4 \end{bmatrix}$$

$$= \frac{1}{\Delta} \left\{ \frac{1}{s} \left[R_2R_3Z_4 \, (e^{r3} - e^{r2}) + R_2R_4Z_3(e^{r2} - e^{r4}) + R_3R_4Z_2(e^{r4} - e^{r3}) \right] - \left[-R_3R_4(e^{r4} - e^{r3}) + R_2R_4(e^{r4} - e^{r2}) - R_2R_3(e^{r3} - e^{r2}) \right] \right\} \tag{85}$$

$$C_2(s) = \frac{1}{\Delta} \cdot \begin{bmatrix} 1 & \frac{1}{s} & 1 & 1 \\ R_1 & 0 & R_3 & R_4 \\ R_1 e^{r1} & 0 & R_3 e^{r3} & R_4 e^{r4} \\ Z_1 & -1 & Z_3 & Z_4 \end{bmatrix}$$

$$= \frac{1}{\Delta} \left\{ \frac{-1}{s} \left[R_1 R_3 Z_4 (e^{r3} - e^{r1}) + R_1 R_4 Z_3 (e^{r1} - e^{r4}) + R_3 R_4 Z_1 (e^{r4} - e^{r3}) \right] \right.$$

$$\left. + \left[-R_3 R_4 (e^{r4} - e^{r3}) - R_1 R_4 (e^{r1} - e^{r4}) + R_1 R_3 (e^{r1} - e^{r3}) \right] \right\} \quad (86)$$

$$C_3(s) = \frac{1}{\Delta} \cdot \begin{bmatrix} 1 & 1 & \frac{1}{s} & 1 \\ R_1 & R_2 & 0 & R_4 \\ R_1{}^{r1} & R_2 e^{r2} & 0 & R_4 e^{r4} \\ Z_1 & Z_2 & -1 & Z_4 \end{bmatrix}$$

$$= \frac{1}{\Delta} \left\{ \frac{1}{s} \left[R_1 R_2 Z_4 (e^{r2} - e^{r1}) + R_1 R4 Z_2 (e^{r1} - e^{r4}) + R_2 R_4 Z_1 (e^{r4} - e^{r2}) \right] \right.$$

$$\left. + \left[-R_2 R_4 (e^{r2} - e^{r4}) + R_1 R_4 (e^{r1} - e^{r4}) + R_1 R_2 (e^{r2} - e^{r1}) \right] \right\} \quad (87)$$

$$C_4(s) = \frac{1}{\Delta} \cdot \begin{bmatrix} 1 & 1 & 1 & \frac{1}{s} \\ R_1 & R_2 & R_3 & 0 \\ R_1 e^{r1} & R_2 e^{r2} & R_3 e^{r3} & 0 \\ Z_1 & Z_2 & Z_3 & -1 \end{bmatrix}$$

$$= \frac{1}{\Delta} \left\{ \frac{-1}{s} \left[R_1 R_2 Z_3 (e^{r2} - e^{r1}) + R_1 R_3 Z_2 (e^{r1} - e^{r3}) + R_2 R_3 Z_1 (e^{r3} - e^{r2}) \right] \right.$$

$$\left. + \left[-R_2 R_3 (e^{r3} - e^{r2}) + R_1 R_3 (e^{r3} - e^{r1}) - R_1 R_2 (e^{r2} - e^{r1}) \right] \right\} \quad (88)$$

A subroutine may be written for the equations (69), (73), (75), (85-88). Numerical results for temperature profiles may be obtained by using this subroutine with the main program of TEMFLU1. If there are difficulties with the solution to this case, such difficulties probably have their source in the assumed boundary conditions. It seems that the boundary conditions (2) and (3) are not independent.

APPENDIX IV

PROGRAM LISTINGS

1. PROGRAM TEMFLU1

a. PURPOSE:

This program finds the inverse transform of $\bar{v}$ and $\bar{u}$, using Salzer's method of numerical inversion.

b. USAGE:

(1) INPUT FORMATS

The input data are read from two cards. The first card reads 8 parameters in floating point format 8F10.5. The second card reads 3 parameters in floating point format 3F10.5 and the run number M is fixed point format I3:

TKS	=	Solid thermal conductivity,	Btu/hr ft^2 ^oF/ft
TKW	=	Fluid thermal conductivity,	"
ROS	=	Density of solid phase,	lb mass/ft^3
ROW	=	Density of fluid phase,	lb mass/ft^3
CS	=	Specific heat of solid phase,	Btu/lb mass ^oF
CW	=	Specific heat of fluid phase,	"
POR	=	Porosity of porous media,	dimensionless
HA	=	Heat transfer coefficient, based on a unit volume of bulk porous media,	Btu/hr ft^3 ^oF
VEL	=	Fluid interstitial velocity,	ft/hr
X1	=	Distance from point of fluid injection,	ft
F	=	Number of time units per hour,	time units/hr
M	=	Run number - Set M=0 on last data card to stop the program	

(2) OUTPUT FORMATS

A = Ratio of thermal diffusivities, dimensionless

B = Ratio of thermal conductivities, "

C = Dimensionless parameter

Y = Dimensionless distance

T = Dimensionless time

N = Order of polynomial

V = Fluid temperature fraction

U = Solid temperature fraction

ERV = Difference between two values of V using two adjacent orders of polynomial

ERU = Difference between two values of U using two adjacent orders of polynomial

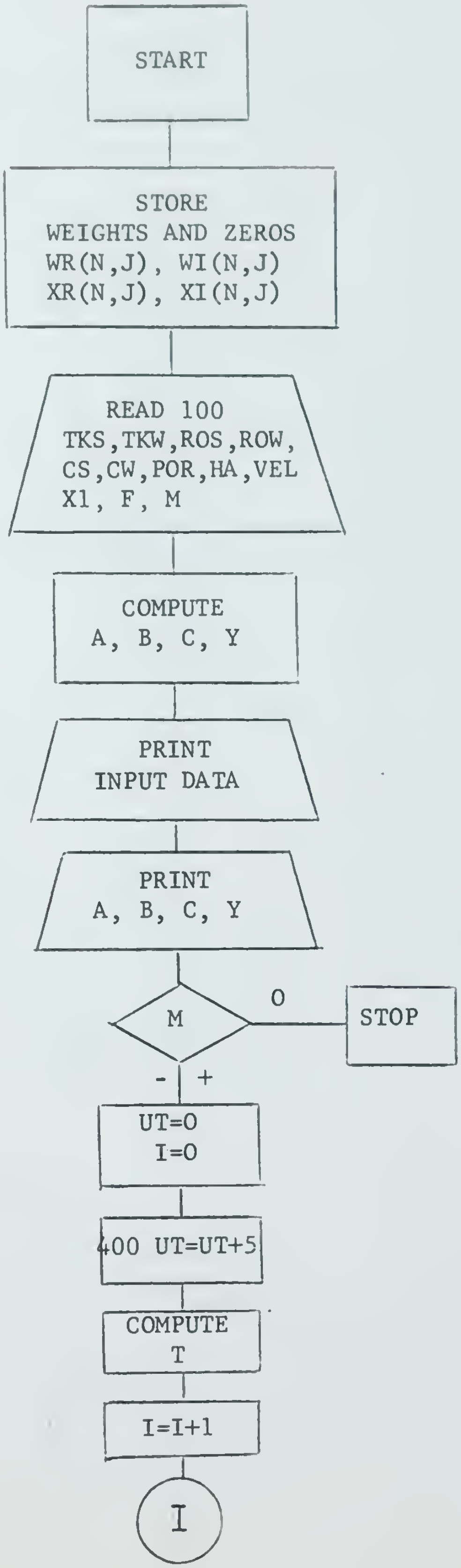
c. SPECIAL INSTRUCTIONS

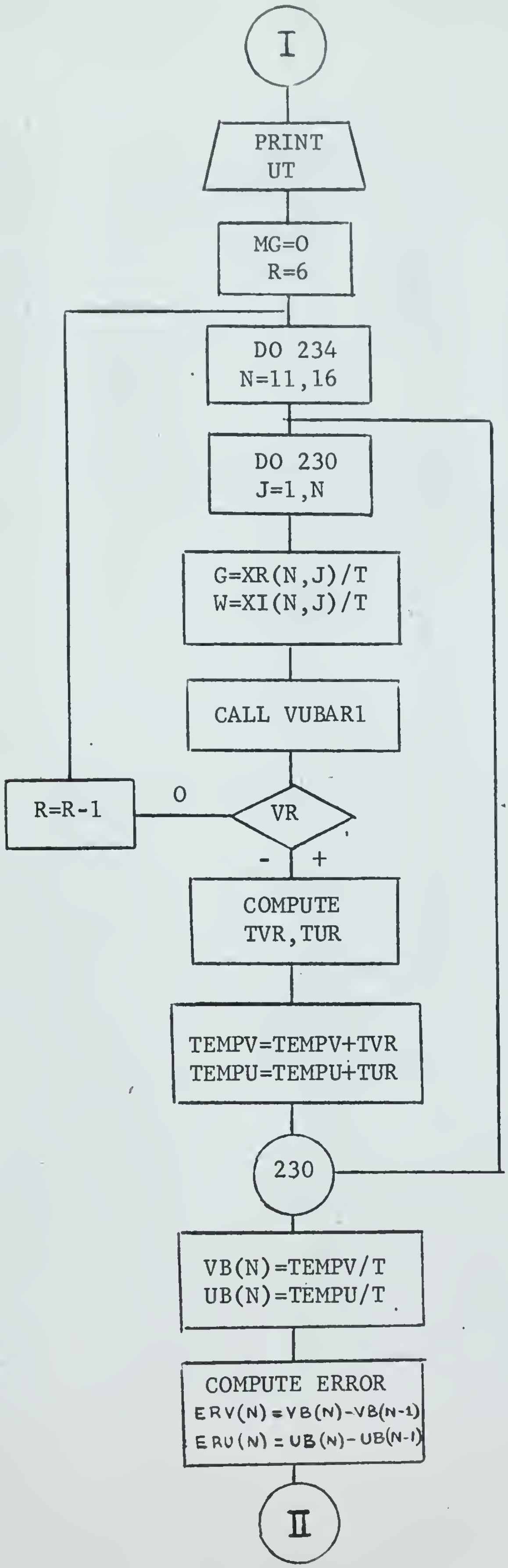
The program TEMFLU1 calls the subroutine VUBAR1. It also uses the function AITKENF to interpolate V and U.

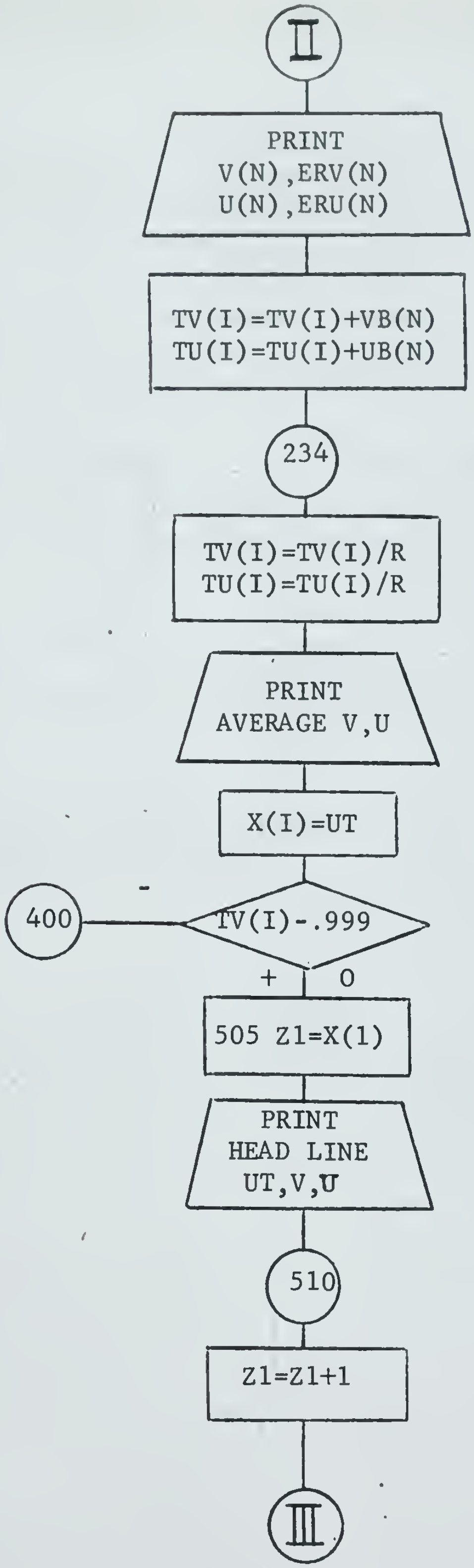
d. MATHEMATICAL METHOD

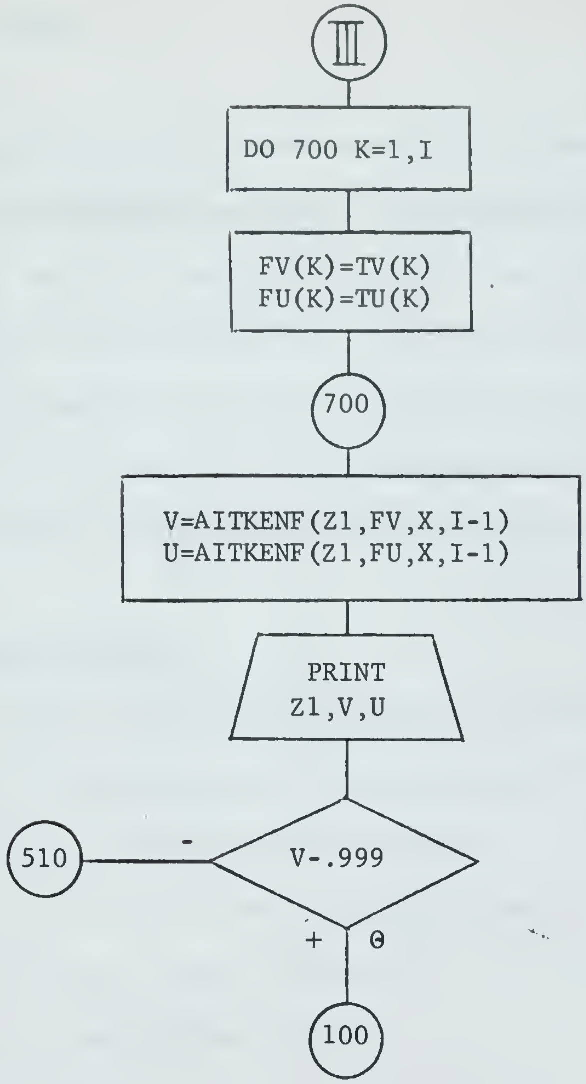
See section 3 and 4 of this thesis.

PROGRAM TEMLU1




I
PRINT
UT
MG=0
R=6
DO 234
N=11,16
DO 230
J=1,N
G=XR(N,J)/T
W=XI(N,J)/T
CALL VUBAR1
R=R-1
0
VR
-
+
COMPUTE
TVR,TUR
TEMPV=TEMPV+TVR
TEMPU=TEMPU+TUR
230
VB(N)=TEMPV/T
UB(N)=TEMPU/T
COMPUTE ERROR
ERV(N)=VB(N)-VB(N-1)
ERU(N)=UB(N)-UB(N-1)
II


II
PRINT
V(N),ERV(N)
U(N),ERU(N)
TV(I)=TV(I)+VB(N)
TU(I)=TU(I)+UB(N)
234
TV(I)=TV(I)/R
TU(I)=TU(I)/R
PRINT
AVERAGE V,U
X(I)=UT
400
-
TV(I)-.999
+
0
505 Z1=X(1)
PRINT
HEAD LINE
UT,V,U
510
Z1=Z1+1
III


III
DO 700 K=1,I
FV(K)=TV(K)
FU(K)=TU(K)
700
V=AITKENF(Z1,FV,X,I-1)
U=AITKENF(Z1,FU,X,I-1)
PRINT
Z1,V,U
510
-
V-.999
+
0
100

2. SUBROUTINE VUBAR1

a. PURPOSE:

This subroutine, given the values of the dimensionless parameters and of the transformed variable S, calculates the complex coefficients of the quartic equation (18), calls the subroutines POLYRT or COMSUB that find the complex roots of equation (18), calls the subroutine POLYVAL to check the accuracy of the roots, then selects the roots with negative real parts to calculate the coefficients C_1 and C_2 of equation (19a) and finally computes VR, VI, UR, UI.

b. USAGE:

(1) INPUT ARGUMENTS:

G = Real part of the transformed variable S

W = Imaginary part of the transformed variables

A = Ratio of thermal diffusivities

B = Ratio of thermal conductivities

C = Dimensionless parameter

Y = Dimensionless distance

(2) OUTPUT FORMATS:

(a) "N = ,J = ,MG = , PZR OR PZI IS LARGER THAN 1.E-4" is printed if the roots are not accurate. "N" is the order of polynomial; "J" identifies one of the zeros of this order; "PZR" and "PZI" are the values of the quartic equation (18) evaluated at the root. "MG" refers to the subroutine used for solving the quartic equation; "MG = 0" or "MG = 3" is printed if POLYRT has been used; "MG = 1" or "MG = 2" is printed if COMSUB has been used; "MG = 4" or "MG = 5" is printed if both subroutines have been used; the number is 4 if POLYRT has been used first, and 5 if COMSUB has been used first.

(b) "N = ,J = ,MG = , ONE ROOT HAS NEGATIVE REAL PART" is printed if only one root with negative real part has been found.

(c) "N = ,J = ,MG = , THREE ROOTS HAVE NEGATIVE REAL PART" is printed if three roots with negative real part have been found.

(d) "N = ,J = ,MG = , CONSTANT VECTOR NOT EQUAL TO CHECK VECTOR" is printed if the calculation of C_1 and C_2 has not been accurate.

(e) VR = Real part of the transform of v

(f) VI = Imaginary part of the transform of v

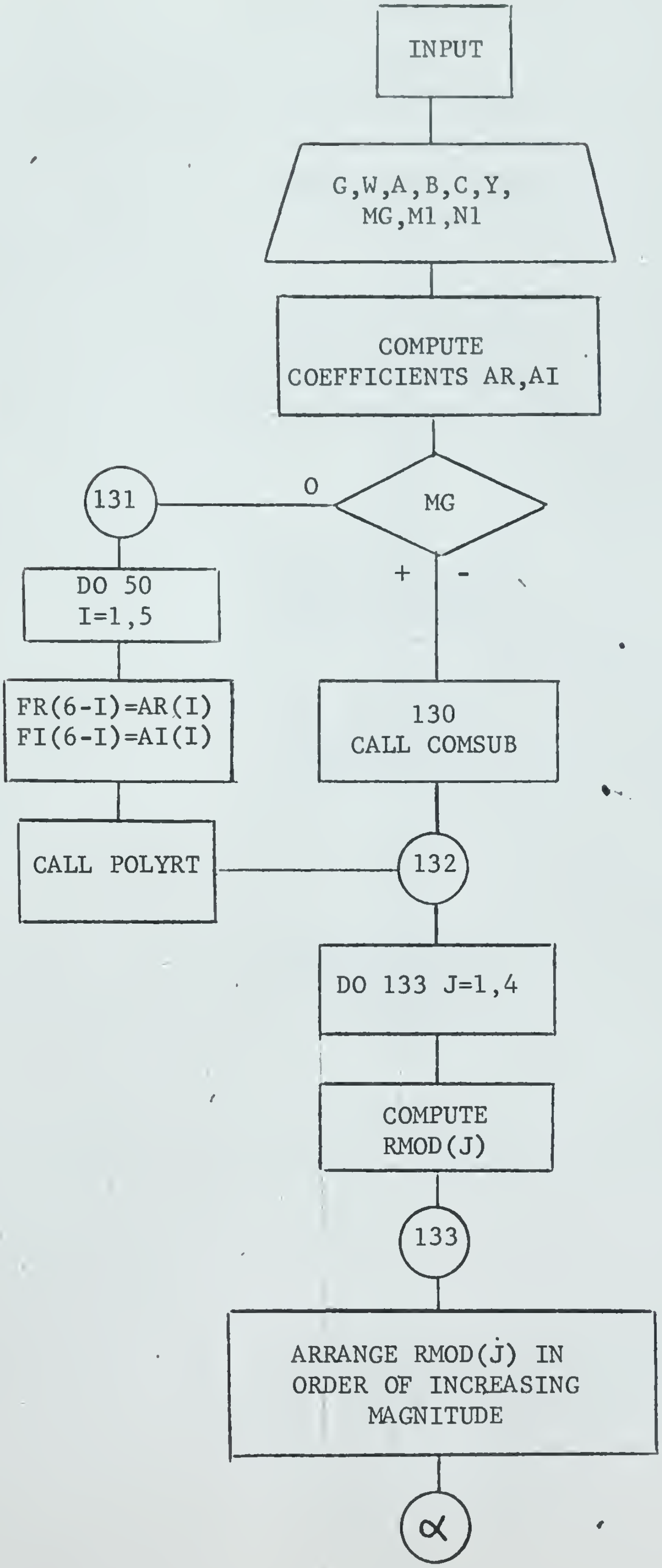
(g) UR = Real part of the transform of u

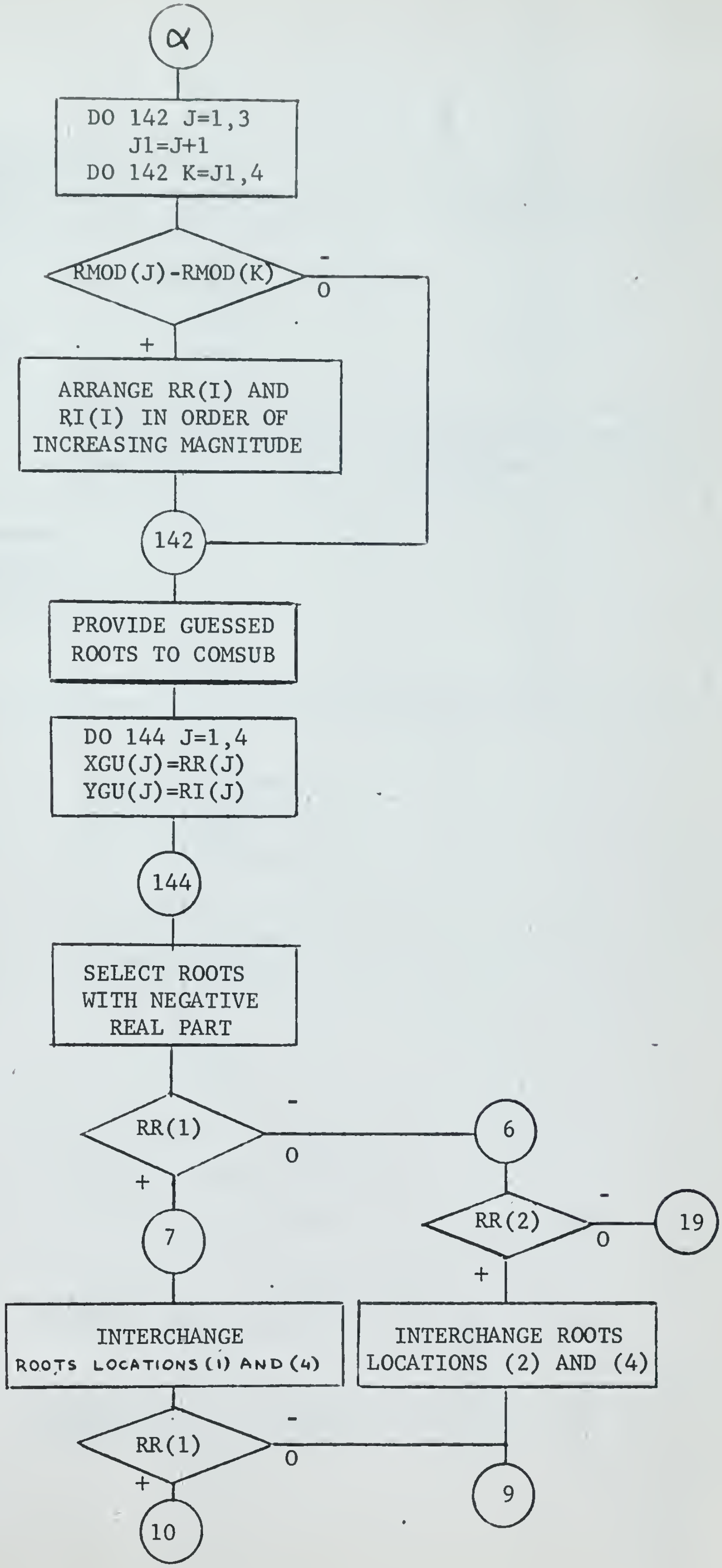
(h) UI = Imaginary part of the transform of u

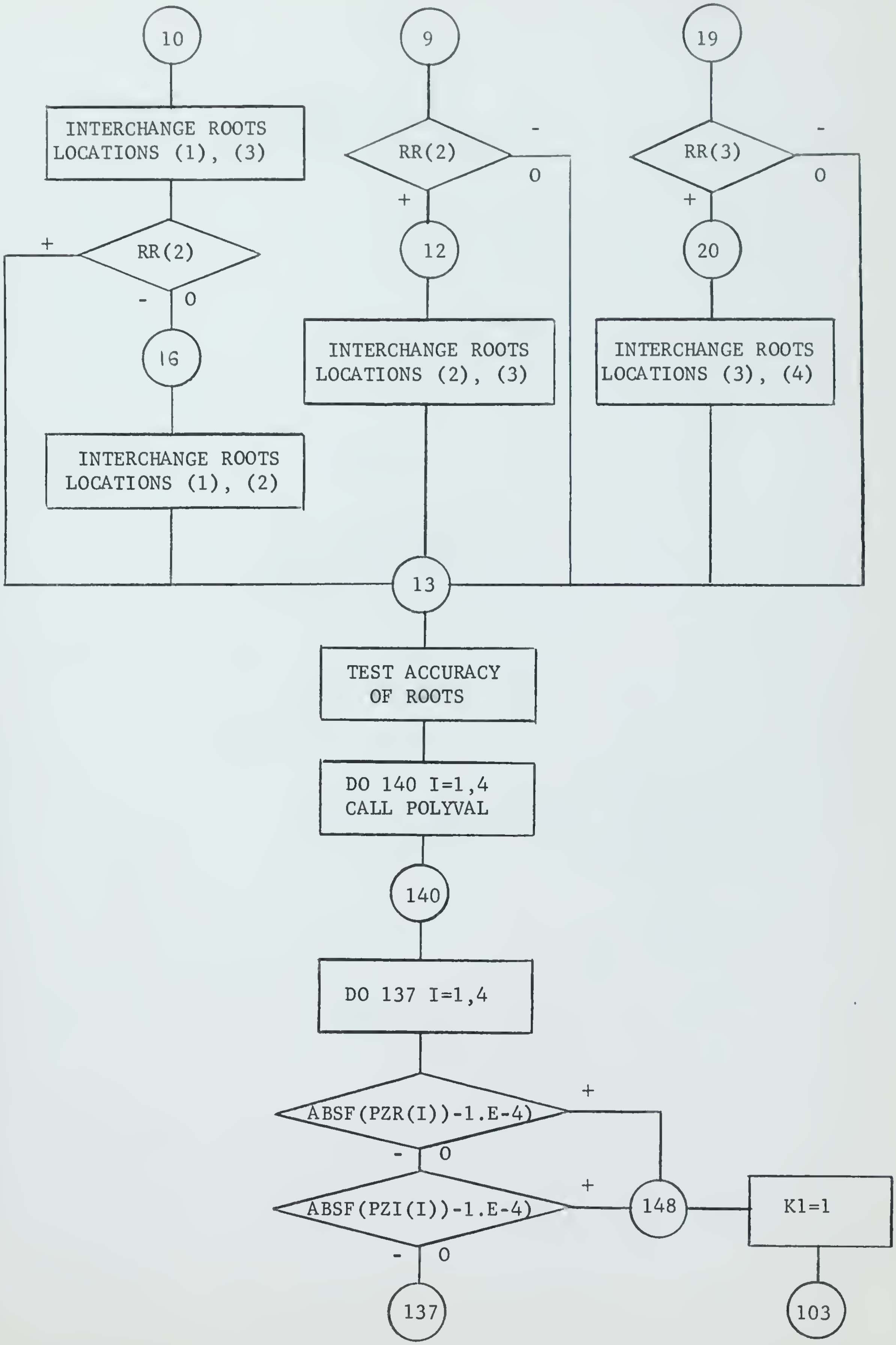
(i) VR, VI, UR and VI are set equal to zero if one of the outputs (1), (2), (3) or (4) is printed.

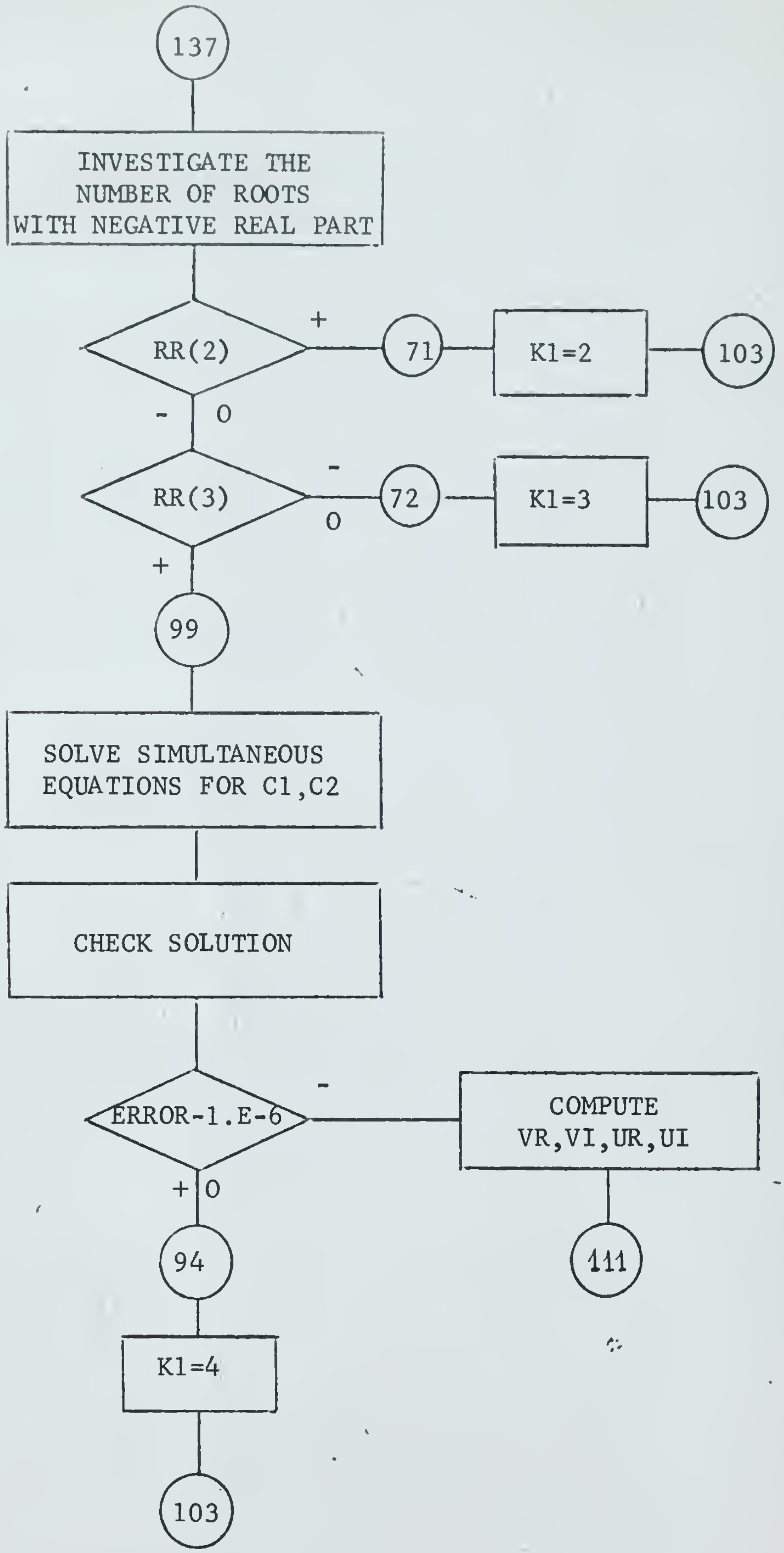
c. <u>SPECIAL INSTRUCTIONS</u>:

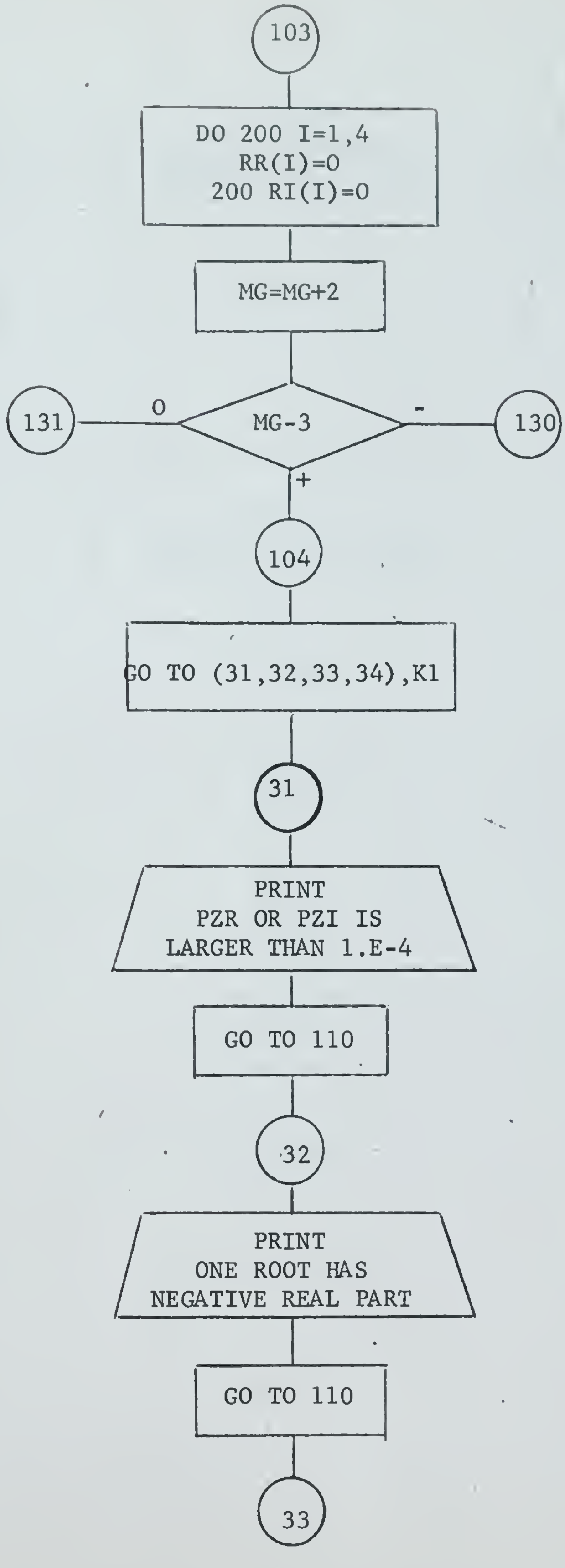
VUBAR1 uses the subroutine MULT for multiplication of two complex numbers.

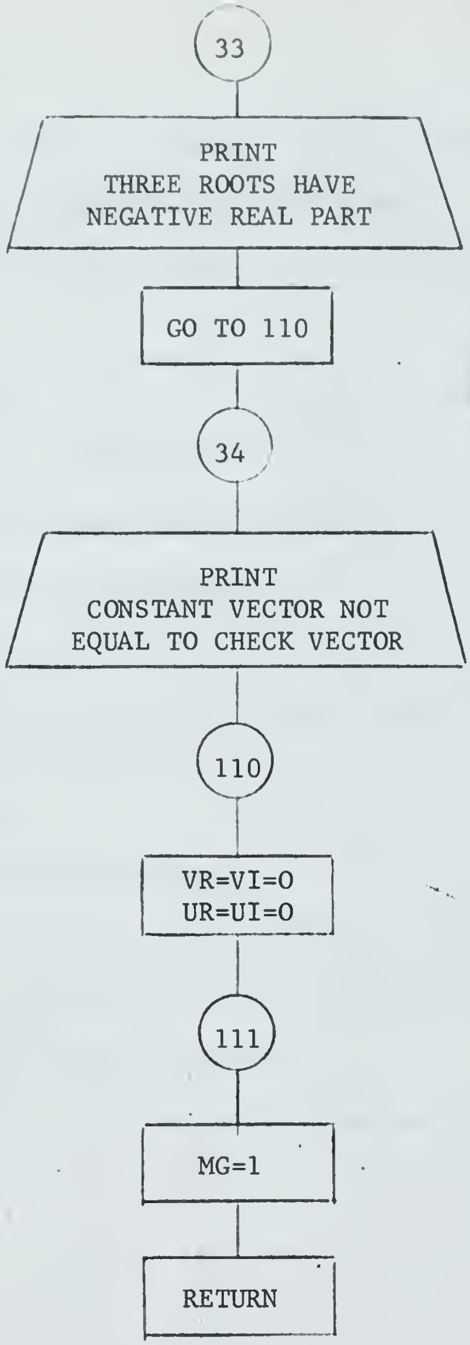

INPUT
G,W,A,B,C,Y,
MG,M1,N1
COMPUTE
COEFFICIENTS AR,AI
MG
0
+
-
131
DO 50
I=1,5
FR(6-I)=AR(I)
FI(6-I)=AI(I)
CALL POLYRT
130
CALL COMSUB
132
DO 133 J=1,4
COMPUTE
RMOD(J)
133
ARRANGE RMOD(J) IN
ORDER OF INCREASING
MAGNITUDE
α


α
DO 142 J=1,3
J1=J+1
DO 142 K=J1,4
RMOD(J)-RMOD(K)
-
0
+
ARRANGE RR(I) AND
RI(I) IN ORDER OF
INCREASING MAGNITUDE
142
PROVIDE GUESSED
ROOTS TO COMSUB
DO 144 J=1,4
XGU(J)=RR(J)
YGU(J)=RI(J)
144
SELECT ROOTS
WITH NEGATIVE
REAL PART
RR(1)
-
0
+
6
7
RR(2)
-
0
+
19
INTERCHANGE
ROOTS LOCATIONS (1) AND (4)
INTERCHANGE ROOTS
LOCATIONS (2) AND (4)
RR(1)
-
0
+
9
10


10
9
19
INTERCHANGE ROOTS LOCATIONS (1), (3)
RR(2)
RR(3)
RR(2)
12
20
16
INTERCHANGE ROOTS LOCATIONS (2), (3)
INTERCHANGE ROOTS LOCATIONS (3), (4)
INTERCHANGE ROOTS LOCATIONS (1), (2)
13
TEST ACCURACY OF ROOTS
DO 140 I=1,4
CALL POLYVAL
140
DO 137 I=1,4
ABSF(PZR(I))-1.E-4
ABSF(PZI(I))-1.E-4
148
K1=1
137
103


137
INVESTIGATE THE
NUMBER OF ROOTS
WITH NEGATIVE REAL PART
RR(2)
+
71
K1=2
103
-
0
RR(3)
-
0
72
K1=3
103
+
99
SOLVE SIMULTANEOUS
EQUATIONS FOR C1,C2
CHECK SOLUTION
ERROR-1.E-6
-
COMPUTE
VR,VI,UR,UI
111
+
0
94
K1=4
103


103
DO 200 I=1,4
RR(I)=0
200 RI(I)=0
MG=MG+2
131
0
MG-3
-
130
+
104
GO TO (31,32,33,34),K1
31
PRINT
PZR OR PZI IS
LARGER THAN 1.E-4
GO TO 110
32
PRINT
ONE ROOT HAS
NEGATIVE REAL PART
GO TO 110
33


33
PRINT
THREE ROOTS HAVE
NEGATIVE REAL PART
GO TO 110
34
PRINT
CONSTANT VECTOR NOT
EQUAL TO CHECK VECTOR
110
VR=VI=0
UR=UI=0
111
MG=1
RETURN

3. PROGRAM JENKINS

a. PURPOSE:

This program finds the solution to the special case where the fluid and solid temperature are assumed to be equal. The equation (58) is programmed, using the function ERFN which calculates the error function.

b. INPUT FORMATS:

A	=	Ratio of thermal diffusivities,	dimensionless
B	=	Ratio of thermal conductivities,	"
C	=	Dimensionless parameter ,	"
T	=	Dimensionless time	"
M	=	Run number. Set M = 0 on last data card to stop the program.	

c. OUTPUT FORMATS

X = Dimensionless distance

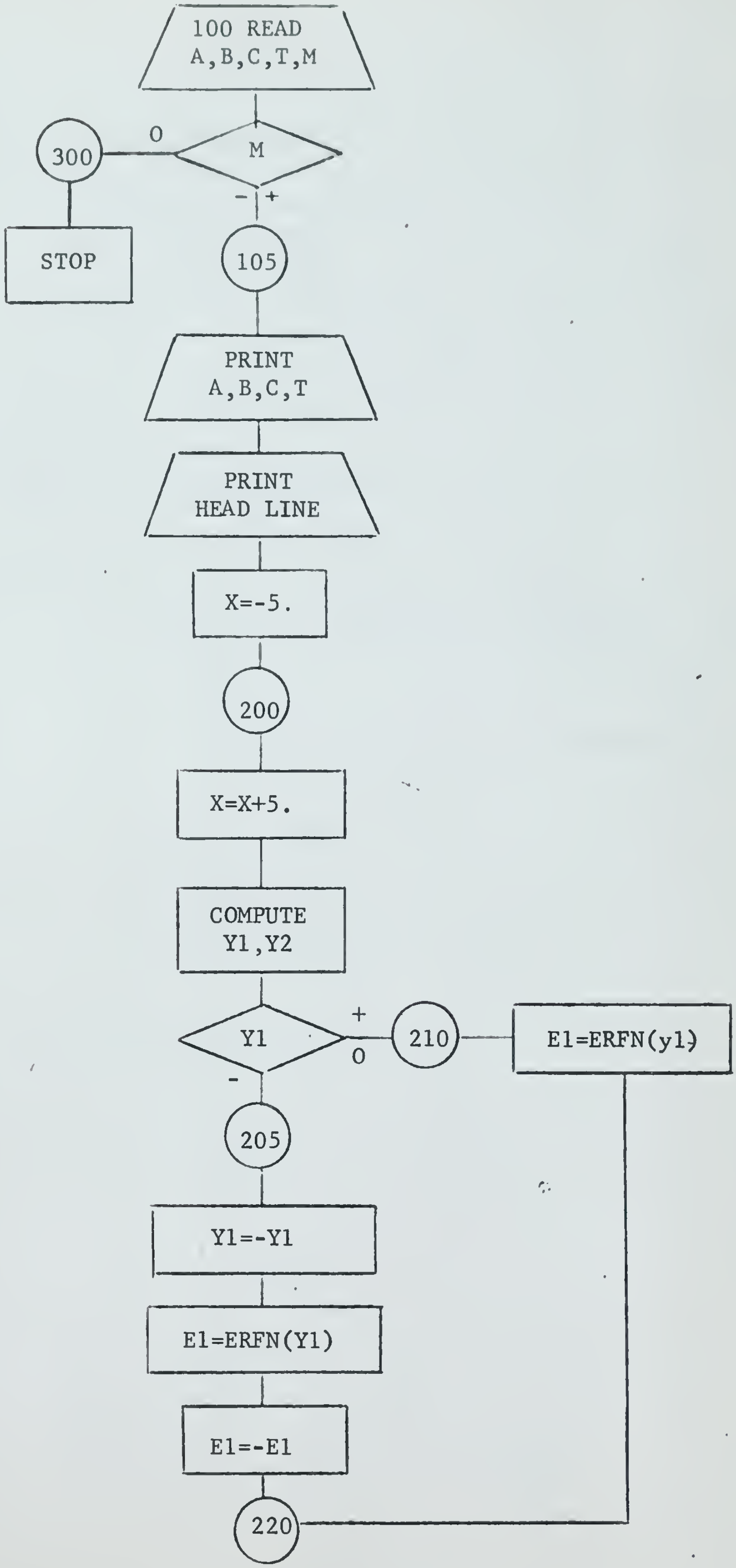
ERC = Value of the first term of equation (58)

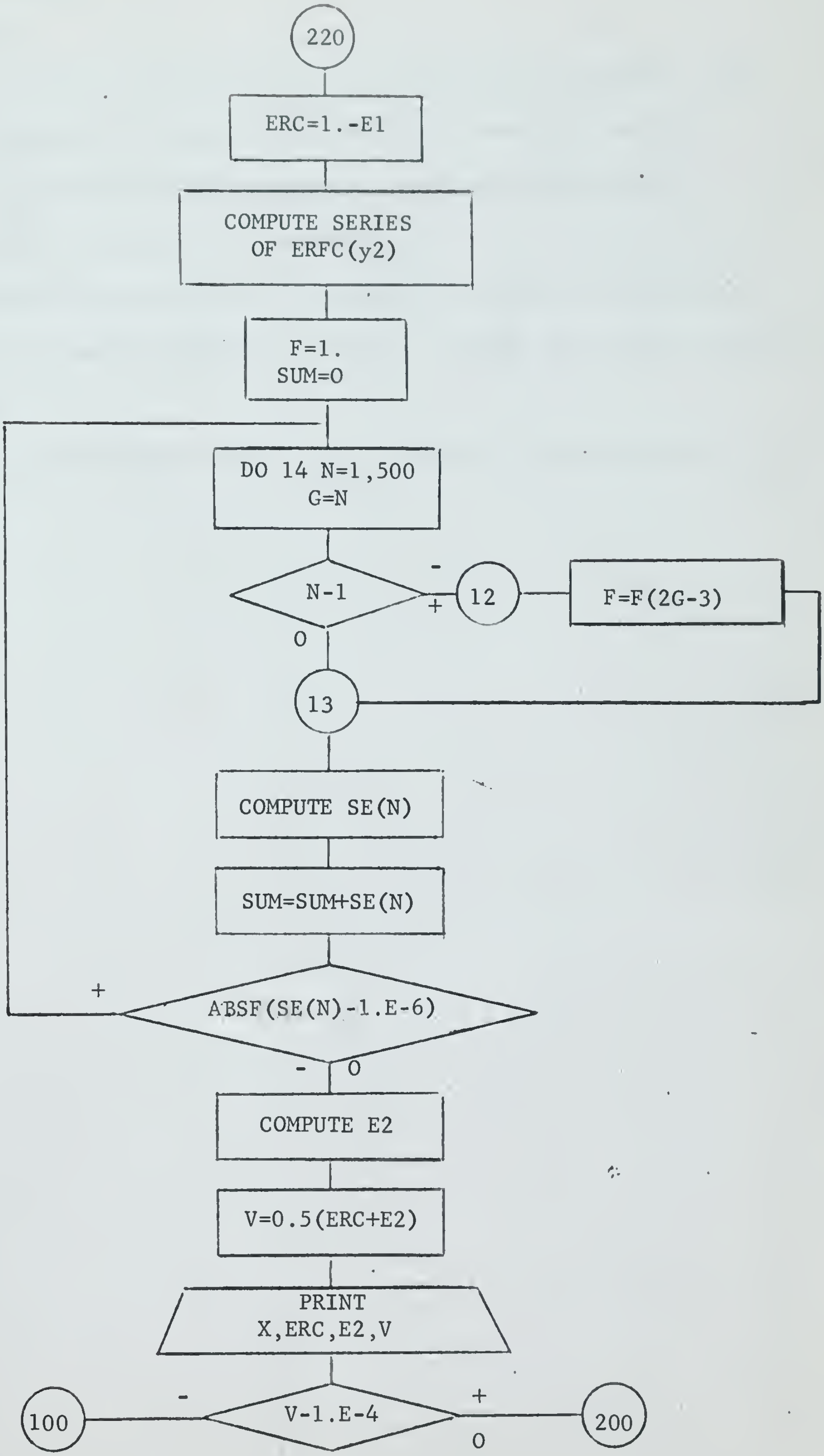
E2 = Value of the second term of equation (58)

V = Fluid temperature fraction

ERC and E2 are printed out to show their relative importance.

PROGRAM JENKINS




220
ERC=1.-E1
COMPUTE SERIES OF ERFC(y2)
F=1.
SUM=0
DO 14 N=1,500
G=N
N-1
-
+
0
12
F=F(2G-3)
13
COMPUTE SE(N)
SUM=SUM+SE(N)
+
ABSF(SE(N)-1.E-6)
-
0
COMPUTE E2
V=0.5(ERC+E2)
PRINT
X,ERC,E2,V
-
+
0
100
V-1.E-4
200

4. PROGRAM SCHUMANN

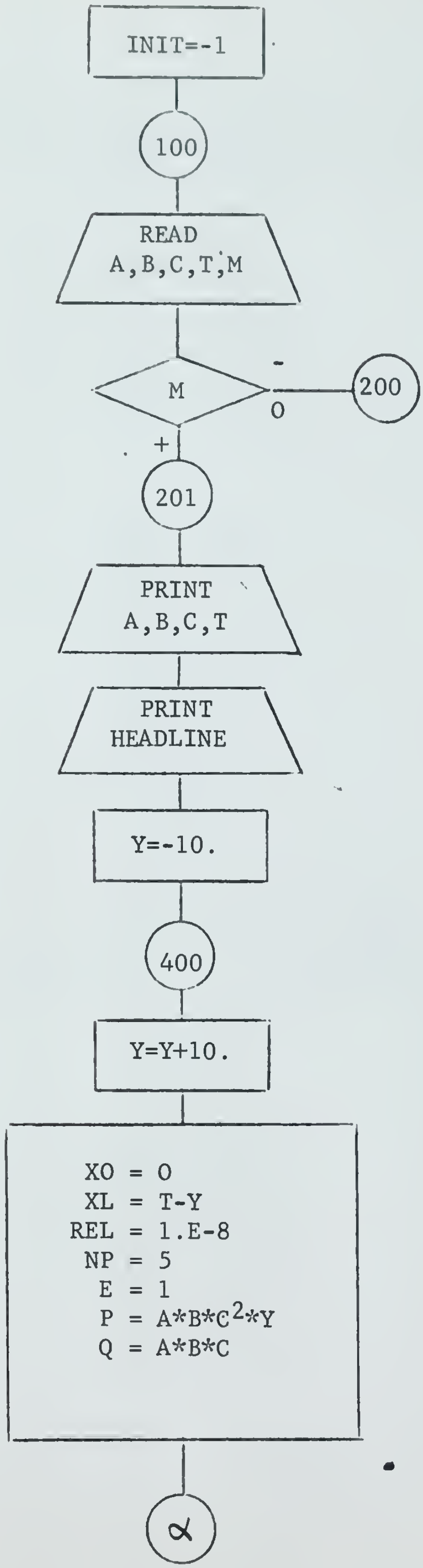
a. PURPOSE:

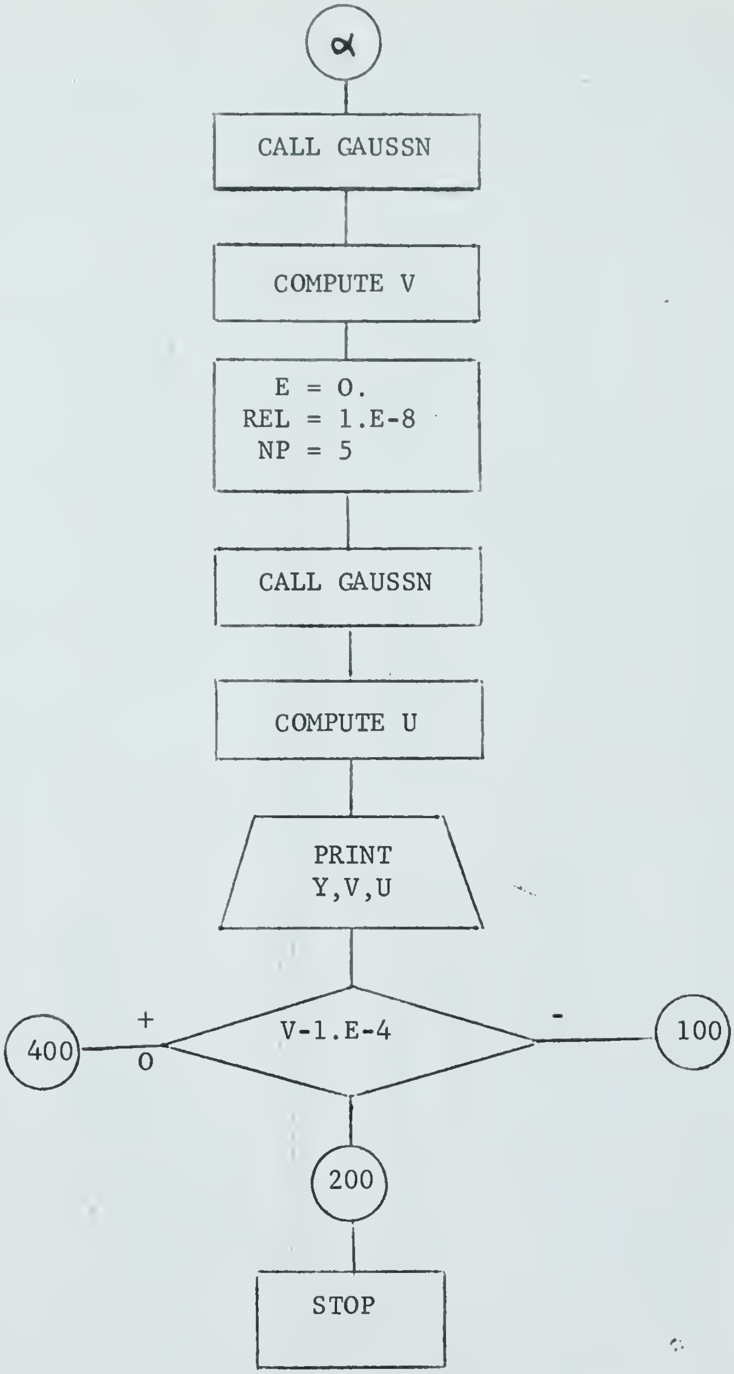
This program finds the solution to the special case where both longitudinal conduction in fluid phase and solid phase are neglected. The equations (42) and (44) are programmed, using the subroutine GAUSSN to evaluate the integrals. GAUSSN itself calls the subroutine FOFX which evaluates the integrands by using the subroutine BESSELL to find the values of modified BESSEL functions of first kind (order 0 and 1).

b. USAGE:

The input and output format are the same as those defined in program Jenkins.

PROGRAM SCHUMANN




α
CALL GAUSSN
COMPUTE V
E = 0.
REL = 1.E-8
NP = 5
CALL GAUSSN
COMPUTE U
PRINT
Y,V,U
V-1.E-4
+
0
-
400
100
200
STOP

```
..JOB115F,HIEP TEMFLU1
      PROGRAM TEMFLU1

C GENERAL CASE WHERE BOTH CONDUCTION AND CONVECTION ARE CONSIDERED
C DIMENSIONLESS TEMPERATURE V.S DIMENSIONLESS TIME
C DIMENSIONLESS DISTANCE FROM 0 TO INFINITY

C TKS IS PSEUDO SOLID CONDUCTIVITY
C TKW IS PSEUDO FLUID CONDUCTIVITY
C CS IS SOLID SPECIFIC HEAT
C CW IS FLUID SPECIFIC HEAT
C ROS IS SOLID DENSITY
C ROW IS FLUID DENSITY
C POR IS POROSITY
C HA IS HEAT TRANSFER COEFFICIENT
C VEL IS VELOCITY
C X1 IS DISTANCE IN FEET
C Y IS DIMENSIONLESS DISTANCE
C UT IS TIME UNIT
C F IS NUMBER OF TIME UNIT PER HOUR

C A IS RATIO OF THERMAL DIFFUSIVITIES
C B IS RATIO OF THERMAL CONDUCTIVITIES
C C IS DIMENSIONLESS PARAMETER LAMBDA
C T IS DIMENSIONLESS TIME
C M IS RUN NUMBER
C SET M = 0 ON LAST DATA CARD
C USING SALZER,S METHOD FOR INVERTING LAPLACE TRANSFORMS
C S= G+IW

      DIMENSION XR(20,20),XI(20,20),WR(20,20),WI(20,20),VB(20),UB(20),
     1FRV(20),FRU(20),X(20),FV(20),FU(20),TV(20),TU(20),Z(200),
     2V(200),U(200)

      XR(11,1)=+.054670344380661E+2
      XR(11,2)=XR(11,1)
```

```
XR(11,3)=+.09235954040419E+2
XR(11,4)=XR(11,3)
XR(11,5)=+.116029782674372E+2
XR(11,6)=XR(11,5)
XR(11,7)=+.131123697248751E+2
XR(11,8)=XR(11,7)
XR(11,9)=+.139626435483486E+2
XR(11,10)=XR(11,9)
XR(11,11)=+.142380399544621E+2
XI(11,1)=+.176032980318069E+2
XI(11,2)=-XI(11,1)
XI(11,3)=+.137187257141666E+2
XI(11,4)=-XI(11,3)
XI(11,5)=+.101548327984373E+2
XI(11,6)=-XI(11,5)
XI(11,7)=+.067205058221876E+2
XI(11,8)=-XI(11,7)
XI(11,9)=+.033474764181901E+2
XI(11,10)=-XI(11,9)
XI(11,11)=0.0
WR(11,1)=+.226353719378214E3
WR(11,2)=WR(11,1)
WR(11,3)=-.192135360830204E+4
WR(11,4)=WR(11,3)
WR(11,5)=-.232447875840433E+5
WR(11,6)=WR(11,5)
WR(11,7)=+.203708932399208E+6
WR(11,8)=WR(11,7)
WR(11,9)=-.584733351793539E+6
WR(11,10)=WR(11,9)
WR(11,11)=+.811939413734596E+6
WI(11,1)=-.091337824489705E+3
WI(11,2)=-WI(11,1)
WI(11,3)=+.741458062287689E+4
WI(11,4)=-WI(11,3)
WI(11,5)=-.649784612524991E+5
```

```
WI(11,6)=-WI(11,5)
WI(11,7)=+.196055170910873E+6
WI(11,8)=-WI(11,7)
WI(11,9)=-.226588957409109E+6
WI(11,10)=-WI(11,9)
WI(11,11)=0.0
XR(12,1)=+.096646029160388E+2
XR(12,2)=XR(12,1)
XR(12,3)=+.122232279801269E+2
XR(12,4)=XR(12,3)
XR(12,5)=+.149894720849361E+2
XR(12,6)=XR(12,5)
XR(12,7)=+.155003991084164E+2
XR(12,8)=XR(12,7)
XR(12,9)=+.056935776058305E+2
XR(12,10)=XR(12,9)
XR(12,11)=+.139287203046514E+2
XR(12,12)=XR(12,11)
XI(12,1)=+.155269887259769E+2
XI(12,2)=-XI(12,1)
XI(12,3)=+.119133708537902E+2
XI(12,4)=-XI(12,3)
XI(12,5)=+.050426730131942E+2
XI(12,6)=-XI(12,5)
XI(12,7)=+.016774090754267E+2
XI(12,8)=-XI(12,7)
XI(12,9)=+.194846293682977E+2
XI(12,10)=-XI(12,9)
XI(12,11)=+.084424969660733E+2
XI(12,12)=-XI(12,11)
WR(12,1)=-.106011986654066E+5
WR(12,2)=WR(12,1)
WR(12,3)=+.131630215740167E+6
WR(12,4)=WR(12,3)
WR(12,5)=+.094173318462161E+7
WR(12,6)=WR(12,5)
```

```
WR(12,7)=-.052191205652078E+7
WR(12,8)=WR(12,7)
WR(12,9)=+.019770417084491E+3
WR(12,10)=WR(12,9)
WR(12,11)=-.540875915592675E+6
WR(12,12)=WR(12,11)
WI(12,1)=-.059947134901648E+5
WI(12,2)=-WI(12,1)
WI(12,3)=-.015802446359525E+6
WI(12,4)=-WI(12,3)
WI(12,5)=-.151555073373935E+7
WI(12,6)=-WI(12,5)
WI(12,7)=+.284094757369523E+7
WI(12,8)=-WI(12,7)
WI(12,9)=+.316226175536523E+3
WI(12,10)=-WI(12,9)
WI(12,11)=+.373470946051152E+6
WI(12,12)=-WI(12,11)
XR(13,1)=+.059071875454784E+2
XR(13,2)=XR(13,1)
XR(13,3)=+.100669707738162E+2
XR(13,4)=XR(13,3)
XR(13,5)=+.128027565656813E+2
XR(13,6)=XR(13,5)
XR(13,7)=+.146872619820812E+2
XR(13,8)=XR(13,7)
XR(13,9)=+.166544961771492E+2
XR(13,10)=XR(13,9)
XR(13,11)=+.159369174838046E+2
XR(13,12)=XR(13,11)
XR(13,13)=+.168888189439782E+2
XI(13,1)=+.213724667907769E+2
XI(13,2)=-XI(13,1)
XI(13,3)=+.173451013895605E+2
XI(13,4)=-XI(13,3)
XI(13,5)=+.136835371252579E+2
```

```
XI(13,6)=-XI(13,5)
XI(13,7)=+.101769443369505F+2
XI(13,8)=-XI(13,7)
XI(13,9)=+.033658144667106E+2
XI(13,10)=-XI(13,9)
XI(13,11)=+.067502384900982F+2
XI(13,12)=-XI(13,11)
XI(13,13)=0.0
WR(13,1)=-.39079716902556E+3
WR(13,2)=WR(13,1)
WR(13,3)=+.13258286039431E+5
WR(13,4)=WR(13,3)
WR(13,5)=-.02702498230006F+6
WR(13,6)=WR(13,5)
WR(13,7)=-.05597138731143F+7
WR(13,8)=WR(13,7)
WR(13,9)=-.86609643353174F+7
WR(13,10)=WR(13,9)
WR(13,11)=+.34509534722402E+7
WR(13,12)=WR(13,11)
WR(13,13)=+.11567777459242F+8
WI(13,1)=-.10553604310534E+3
WI(13,2)=-WI(13,1)
WI(13,3)=-.13341532281686F+5
WI(13,4)=-WI(13,3)
WI(13,5)=+.24252155128905F+6
WI(13,6)=-WI(13,5)
WI(13,7)=-.13281711768448F+7
WI(13,8)=-WI(13,7)
WI(13,9)=-.32898467326691F+7
WI(13,10)=-WI(13,9)
WI(13,11)=+.31856321976495E+7
WI(13,12)=-WI(13,11)
WI(13,13)=0.0
XR(14,1)=+.06109537065910 8E+2
XR(14,2)=XR(14,1)
```

```
XR(14,3)=+.104466532469181E+2
XR(14,4)=XR(14,3)
XR(14,5)=+.133474860189496E+2
XR(14,6)=XR(14,5)
XR(14,7)=+.153970406475505E+2
XR(14,8)=XR(14,7)
XR(14,9)=+.168185419175291E+2
XR(14,10)=XR(14,9)
XR(14,11)=+.177208535297203E+2
XR(14,12)=XR(14,11)
XR(14,13)=+.181598875734216E+2
XR(14,14)=XR(14,13)
XI(14,1)=+.232659732506469E+2
XI(14,2)=-XI(14,1)
XI(14,3)=+.191718385658014E+2
XI(14,4)=-XI(14,3)
XI(14,5)=+.154639361328642E+2
XI(14,6)=-XI(14,5)
XI(14,7)=+.119224339983808E+2
XI(14,8)=-XI(14,7)
XI(14,9)=+.084689465826821E+2
XI(14,10)=-XI(14,9)
XI(14,11)=+.050645747484236E+2
XI(14,12)=-XI(14,11)
XI(14,13)=+.016855674473441E+2
XI(14,14)=-XI(14,13)
WR(14,1)=+.28570144704751E+3
WR(14,2)=WR(14,1)
WR(14,3)=+.13950679653728E+5
WR(14,4)=WR(14,3)
WR(14,5)=-.40708888935434E+6
WR(14,6)=WR(14,5)
WR(14,7)=+.29542848615168E+7
WR(14,8)=WR(14,7)
WR(14,9)=-.93440592733119E+7
WR(14,10)=WR(14,9)
```

```
WR(14,11)=+.14168955460489E+8
WR(14,12)=WR(14,11)
WR(14,13)=-.07386335540440E+8
WR(14,14)=WR(14,13)
WI(14,1)=-.42257377031972E+3
WI(14,2)=-WI(14,1)
WI(14,3)=+.24654947254365E+5
WI(14,4)=-WI(14,3)
WI(14,5)=-.14745486220113E+6
WI(14,6)=-WI(14,5)
WI(14,7)=-.05619889208362E+7
WI(14,8)=-WI(14,7)
WI(14,9)=+.68417862739122E+7
WI(14,10)=-WI(14,9)
WI(14,11)=-.23329173881153E+8
WI(14,12)=-WI(14,11)
WI(14,13)=+.40774780001204E+8
WI(14,14)=-WI(14,13)
XR(15,1)=+.06301979854793 3E+2
XR(15,2)=XR(15,1)
XR(15,3)=+.13862078219032 0E+2
XR(15,4)=XR(15,3)
XR(15,5)=+.16065031460803 4E+2
XR(15,6)=XR(15,5)
XR(15,7)=+.17644521765666 4E+2
XR(15,8)=XR(15,7)
XR(15,9)=+.18714332079624 1E+2
XR(15,10)=XR(15,9)
XR(15,11)=+.19335706167276 9E+2
XR(15,12)=XR(15,11)
XR(15,13)=+.10806524913898 0E+2
XR(15,14)=XR(15,13)
XR(15,15)=+.19539651077812 0E+2
XI(15,1)=+.25164472685678 8E+2
XI(15,2)=-XI(15,1)
XI(15,3)=+.17253432587027 1E+2
```

```
XI(15,4)=-XI(15,3)
XI(15,5)=+.13677803043944OE+2
XI(15,6)=-XI(15,5)
XI(15,7)=+.10197743902986 1E+2
XI(15,8)=-XI(15,7)
XI(15,9)=+.067729816593316E+2
XI(15,10)=-XI(15,9)
XI(15,11)=+.033793998819329E+2
XI(15,12)=-XI(15,11)
XI(15,13)=+.210062073041128E+2
XI(15,14)=-XI(15,13)
XI(15,15)=0.0
WR(15,1)=+.38001675351110E+3
WR(15,2)=WR(15,1)
WR(15,3)=+.41388830376509E+6
WR(15,4)=WR(15,3)
WR(15,5)=-.01694097595195E+7
WR(15,6)=WR(15,5)
WR(15,7)=-.11368933115576E+8
WR(15,8)=WR(15,7)
WR(15,9)=+.55740984442647E+8
WR(15,10)=WR(15,9)
WR(15,11)=-.12685729817048E+9
WR(15,12)=WR(15,11)
WR(15,13)=-.40584578578724E+5
WR(15,14)=WR(15,13)
WR(15,15)=+.16456196072199E+9
WI(15,1)=+.50883133061431E+3
WI(15,2)=-WI(15,1)
WI(15,3)=-.61840042872333E+6
WI(15,4)=-WI(15,3)
WI(15,5)=+.60093063354820E+7
WI(15,6)=-WI(15,5)
WI(15,7)=-.24504289234219E+8
WI(15,8)=-WI(15,7)
WI(15,9)=+.49998124803205E+8
```

```
WI(15,10)=-WI(15,9)
WI(15,11)=-.04749121744949E+9
WI(15,12)=-WI(15,11)
WI(15,13)=+.09752029122456E+5
WI(15,14)=-WI(15,13)
WI(15,15)=0.0
XR(16,1)=+.14350276293898 5E+2
XR(16,2)=XR(16,1)
XR(16,3)=+.11148923555154 4E+2
XR(16,4)=XR(16,3)
XR(16,5)=+.16696741637279 4E+2
XR(16,6)=XR(16,5)
XR(16,7)=+.18422718844967 5E+2
XR(16,8)=XR(16,7)
XR(16,9)=+.19646097429403 3E+2
XR(16,10)=XR(16,9)
XR(16,11)=+.20432297698379 8E+2
XR(16,12)=XR(16,11)
XR(16,13)=+.20817316216422 4E+2
XR(16,14)=XR(16,13)
XR(16,15)=+.06485628324494 8E+2
XR(16,16)=XR(16,15)
XI(16,1)=+.19051087358918 0E+2
XI(16,2)=-XI(16,1)
XI(16,3)=+.22847389503912 4E+2
XI(16,4)=-XI(16,3)
XI(16,5)=+.15442080892659 5E+2
XI(16,6)=-XI(16,5)
XI(16,7)=+.11935724977767 5E+2
XI(16,8)=-XI(16,7)
XI(16,9)=+.08490344494121 9E+2
XI(16,10)=-XI(16,9)
XI(16,11)=+.05081295339899 8E+2
XI(16,12)=-XI(16,11)
XI(16,13)=+.01691716342881 6E+2
XI(16,14)=-XI(16,13)
```

```
      XI(16,15)=+.270674101802452E+2
      XI(16,16)=-XI(16,15)
      WR(16,1)=+.832343312083 7E+6
      WR(16,2)=WR(16,1)
      WR(16,3)=+.0291507593847E+5
      WR(16,4)=WR(16,3)
      WR(16,5)=-.112187255804 6E+8
      WR(16,6)=WR(16,5)
      WR(16,7)=+.5843963892001E+8
      WR(16,8)=WR(16,7)
      WR(16,9)=-.1537399707302E+9
      WR(16,10)=WR(16,9)
      WR(16,11)=+.2102572434385E+9
      WR(16,12)=WR(16,11)
      WR(16,13)=-.1045727057607E+9
      WR(16,14)=WR(16,13)
      WR(16,15)=-.7466751219346E+3
      WR(16,16)=WR(16,15)
      WI(16,1)=+.9239995259706E+6
      WI(16,2)=-WI(16,1)
      WI(16,3)=-.6025331421497E+5
      WI(16,4)=-WI(16,3)
      WI(16,5)=-.0285904207613E+8
      WI(16,6)=-WI(16,5)
      WI(16,7)=-.1382716922874E+8
      WI(16,8)=-WI(16,7)
      WI(16,9)=+.1171501818490E+9
      WI(16,10)=-WI(16,9)
      WI(16,11)=-.3520092325588E+9
      WI(16,12)=-WI(16,11)
      WI(16,13)=+.5834154653451E+9
      WI(16,14)=-WI(16,13)
      WI(16,15)=+.2334187148757E+3
      WI(16,16)=-WI(16,15)

  100 READ 101,TKS,TKW,ROS,ROW,CS,CW,POR,HA,VEL,X1,F,M
```

```
101 FORMAT(8F10.5/(3F10.5,I3))
    A=(TKS*ROW*CW)/(TKW*ROS*CS)
    B=(TKW*POR)/(TKS*(1.-POR))
    SQ=SQRTF(HA/(TKW*POR))
    C=SQ*(TKW/(ROW*CW*VEL))
    Y=SQ*X1
231 PRINT 232 M
232 FORMAT(//////,52X,13H RUN NUMBER  I3////)
    PRINT 240,TKS,TKW,ROS,ROW,CS,CW
240 FORMAT(//,5HTKS =F10.5,3X,5HTKW =F10.5,3X,5HROS =F10.5,3X,
   15HROW =F10.5,3X,5H CS =F10.5,3X,5H CW =F10.5,3X,/)
    PRINT 250,POR,HA,VEL,X1,F
250 FORMAT(//,5HPOR =F10.5,3X,5H HA =F20.5,3X,5HVEL =F10.5,3X,
   15H X1 =F10.5,5X,5H  F =F10.5)
    PRINT 4,A,B,C,Y
  4 FORMAT(///,5H A = F15.5,5X,5H B = F15.5,5X,5H C = F15.5,5X,
   15H Y = F15.5)
    IF(M)200,200,201
201 UT=0.
202 I=0
400 UT=UT+5.
    I=I+1
    T=SQ*VEL*UT/F
    X(I)=UT
    TV(I)=0.
    TU(I)=0.
    PRINT 10,UT
 10 FORMAT(/////,4H UT=F10.5)
    MG=0
    R=6.
    DO 234 N=11,16
    VB(N)=0.
    UB(N)=0.
    FRV(N)=0.
    FRU(N)=0.
    TEMPV=0.
```

```
      TEMPU=0.
      DO 230 J=1,N
      G=XR(N,J)/T
      W=XI(N,J)/T
      CALL VUBAR1(G,W,A,B,C,Y,VR,VI,UR,UI)

C VR IS SET EQUAL TO ZERO IN SUBROUTINE VUBAR1 IN THE FOLLOWING CASES
C (1) THE ROOT COMPUTATION FAILS,PZR OR PZI LARGER THAN 1.E-6
C (2) ONE OR THREE ROOTS HAVE NEGATIVE REAL PART
C (3) THE CALCULATION OF THE CONSTANTS FAILS

      IF(VR)30,29,30
   29 R=R-1.
      GO TO 234
   30 CALL MULT(VR,VI,WR(N,J),WI(N,J),TVR,TUI,K1)
      CALL MULT(UR,UI,WR(N,J),WI(N,J),TUR,TUI,K1)
      TEMPV=TEMPV+TVR
  230 TEMPU=TEMPU+TUR
      VB(N)=TEMPV/T
      UB(N)=TEMPU/T
  280 ERV(N)=VB(N)-VB(N-1)
      ERU(N)=UB(N)-UB(N-1)
      TV(I)=TV(I)+VB(N)
      TU(I)=TU(I)+UB(N)
      PRINT 235(N,VB(N),ERV(N),UB(N),ERU(N))
  235 FORMAT(/,5X,3H N=I2,5X,4H V =F15.9,5X,6H ERV =F15.9,5X,
     14H U =F15.9,5X,6H ERU =F15.9)
  234 CONTINUE

C CALCULATE THE AVERAGE OF V AND U

      TV(I)=TV(I)/R
      TU(I)=TU(I)/R
      PRINT 500,UT,TV(I),TU(I)
  500 FORMAT(//,4HUT =F10.5,5X,3HV =F15.9,5X,3HU =F15.9)
      IF(TV(I)-.999)400,505,505
```

```
505 Z1=X(1)
    PRINT 515
515 FORMAT(////,39X,2HUT,16X,1HV,19X,1HU,///)
510 J=J+1
    Z1=Z1+1.
    V(J)=0.
    U(J)=0.
    DO 700 K=1,I

C SAVE TV AND TU FOR FV AND FU ARE DESTROYED BY AITKENF FUNCTION
C INTERPOLATION BY AITKEN,S METHOD

    FV(K)=TV(K)
700 FU(K)=TU(K)
    V(J)=AITKENF(Z1,FV,X,I-1)
    U(J)=AITKENF(Z1,FU,X,I-1)
    Z(J)=Z1
    PRINT 520,Z(J),V(J),U(J)
520 FORMAT(/,34X,E10.5,7X,E15.9,6X,E15.9)
    IF(U(J)-.999)510,100,100
200 STOP
    END



    MACHINE AITKENF(Z, FX, X, NTH)
    CON(E1=6171232065502700B,E2=6171232045131200B)
    LOC(ERP=24)
1N   SLJ (*)        SIL 1 (1RS)     .EXIT/ENTRY
1Z   SIU 2 (2RS)    FNI 1 (*)       . ADDRESS OF Z
2FX  SIU 1 (1ZX)    FNA   (*)       . ADDRESS OF FX
3X   SIU 1 (2ZX)    FNI 2 (*)       . ADDRESS OF X
4NTH INA (1)        LIL 1 (*)       . ADDRESS OF NTH
     SAU (1F)       SAL (3F)        .
     INA (1)        SAU (2F)        .
     INI 2 (1)      SIL 2 (1LP)     .
```

```
      SIL 2 (2ZX)     INI 2 (1)       .
      SIU 2 (1LP)     SIL 2 (1ZX)     .
      FNA 1 (N)       AJP (2ERR)      . TEST FOR NTH EQUAL ZERO
      SIU 1 (1MN)     INI 1 (-1)      .
1MN   FNI 2 (*)       INI 2 (-1)      . .
1LP   LDA 1 (*)       FSB 2 (*)       .  .  DO I = N:1,1
      AJP (1ERR)      STA (DENOM)     .   . DO J = I,1
1ZX   LDA (*)         FSB 1 (*)       .  .
1F    FMU 2 (*)       STA (SUBT)      . ...   ((Z-X(J))*F(I) - (Z-X(I))*F
2ZX   LDA (*)         FSB 2 (*)       .  .    )-X(J)
2F    FMU 1 (*)       FSB (SUBT)      .   .
3F    FDV (DENOM)     STA 2 (*)       .  .
      IJP 2 (1LP)     SIU 1 (1MN)     . .
1RS   IJP 1 (1MN)     FNI 1 (*)       . RESTORE
2RS   FNI 2 (*)       SLJ (1N)        .RESTORE
1ERR  FNA (F1)        SLJ 4 (ERP)     . ERROR JUMP ON NTH ZERO
2ERR  FNA (E2)          SLJ 4 (ERP)   . ERROR JUMP ON X(I) =X(J)
      END

       SUBROUTINE   DIVD(XR,XI,YR,YI,ZR,ZI,KER)
       CALL PROD(XR,XI,YR,-YI,B1,B2,PR,PI,DR,DI)
          LDA(B2)   AJP1(1)   ENA(3)    SLJ(3)                       .
    2     ENA(2)    SLJ(3)                                           .
    1  T=DR*DR+DI*DI
          LDA(B1)  -FDV(B2)  +EXF7(141B)SLJ(2)    STA(B1)
          LDA(PR)   FDV(T)   -FMU(B1)  +FXF7(141B)SLJ(2)    STA(ZR)
          LDA(PI)   FDV(T)   -FMU(B1)  +FXF7(141B)SLJ(2)   STA(ZI)ENA(1)
    3     STA(KER)                                                   .
       END
       SUBROUTINE   MULT(XR,XI,YR,YI,ZR,ZI,KER)
       CALL PROD(XR,XI,YR,YI,B1,B2,PR,PI,D1,D2)
          LDA(B2)  -FMU(B1)  +EXF7(141B)SLJ(1)    STA(B1)            .
          LDA(PR)  -FMU(B1)  +EXF7(141B)SLJ(1)    STA(ZR)            .
          LDA(PI)  -FMU(B1)  +EXF7(141B)SLJ(1)    STA(ZI)
          ENA(1)    STA(KER)  SLJ(L+2)                               .
    1     ENA(2)    STA(KER)
```

```
      END
      SUBROUTINE   PROD(XR,XI,YR,YI,B1,B2,PR,PI,DR,DI)
      CALL NORM(XR,XI,B1,AR,AI)
      CALL NORM(YR,YI,B2,DR,DI)
      PR=AR*DR-AI*DI
      PI=AI*DR+AR*DI
      END
      SUBROUTINE   NORM(A1,A2,B1,S1,S2)
 1A       SLJ(1)     +SEV7(70000B)         ZRO(0)     +ZRO(4000B)ZRO(0)
 1        LDA(1A+1)  LDQ(A1)    QJP2(L+1) LQC(A1)    STL(F)     LDQ(A2)
          QJP2(L+1)  LQC(A2)    LDL(1A+1)+THS(E)     SLJ(L+2)   LDA(E)
         +AJP1(L+2)  STA(S1)    STA(B1)   SLJ(L+5)  +ADD(1A+2)  STA(B1)
          LDA(A1)    FDV(B1)    STA(S1)   LDA(A2)    FDV(B1)   +STA(S2)
      END

      SUBROUTINE VUBAR1(MG,N1,M1,G,W,A,B,C,Y,VR,VI,UR,UI)
C EVALUATION OF MAGNITUDE OF COMPLEX V(Y,S)
C S IS LAPLACE TRANSFORM OPERATOR
      DIMENSION AR(50),AI(50),FR(50),FI(50),RR(50),RI(50),PZR(4),PZI(4),
     1RMOD(4),XGU(4),YGU(4)
      COMMON MG,M1,N1
      N=4
      NRLCMP=1
      NPREC=2
      AR(1)=A*C
      AR(2)=-A
      AR(3)=-(A+1.0)*G-A*C*(B+1.0)
      AR(4)=A*B+G/C
      AR(5)=(G*G-W*W)/C+(A*B+1.0)*G
      AI(1)=0
      AI(2)=0
      AI(3)=-(A+1.0)*W
      AI(4)=W/C
      AI(5)=(2.0*G*W)/C+(A*B+1.0)*W

C SELECT POLYRT OR COMSUB
```

```
      IF(MG)130,131,130

C SAVE AR(I) BECAUSE FR(I) ARE DESTROYED IN POLYRT

  131 DO 50 I=1,5
      FR(6-I)=AR(I)
   50 FI(6-I)=AI(I)
      DELTA2=1.0E-6
      CALL POLYRT(FR,FI,N,RR,RI,DELTA2)
      GO TO 132
  130 CALL COMSUB(N,MG,AR,AI,RR,RI,IND,XGU,YGU)

C ROOTS ARE ARRANGED IN INCREASING MODULUS TO PROVIDE GUESSED ROOTS TO C

  132 DO 133 J=1,4
  133 RMOD(J)=SQRTF(RR(J)**2+RI(J)**2)
      DO 142 J=1,3
      J1=J+1
      DO 142 K =J1,4
      IF(RMOD(J)-RMOD(K))142,142,143
  143 TEMP=RMOD(J)
      RMOD(J)=RMOD(K)
      RMOD(K)=TEMP
      RRR=RR(J)
      RII=RI(J)
      RR(J)=RR(K)
      RI(J)=RI(K)
      RR(K)=RRR
      RI(K)=RII
  142 CONTINUE
      DO 144 J=1,4

C INVESTIGATE SIGN OF REAL PART
C ONLY ROOTS WITH NEGATIVE REAL PART ARE USED
```

```
      XGU(J)=RR(J)
  144 YGU(J)=RI(J)
      IF(RR(1))6,6,7
    7 TEMPR=RR(1)
      TEMPI=RI(1)
      RR(1)=RR(4)
      RI(1)=RI(4)
      RR(4)=TEMPR
      RI(4)=TEMPI
    8 IF(RR(1))9,9,10
   10 TEMPR=RR(1)
      TEMPI=RI(1)
      RR(1)=RR(3)
      RI(1)=RI(3)
      RR(3)=TEMPR
      RI(3)=TEMPI
      IF(RR(2))16,16,13
   16 TEMPR=RR(2)
      TEMPI=RI(2)
      RR(2)=RR(1)
      RI(2)=RI(1)
      RR(1)=TEMPR
      RI(1)=TEMPI
      GO TO 13
    6 IF(RR(2))19,19,11
   11 TEMPR=RR(2)
      TEMPI=RI(2)
      RR(2)=RR(4)
      RI(2)=RI(4)
      RR(4)=TEMPR
      RI(4)=TEMPI
    9 IF(RR(2))13,13,12
   12 TEMPR=RR(2)
      TEMPI=RI(2)
      RR(2)=RR(3)
      RI(2)=RI(3)
```

```
      RR(3)=TEMPR
      RI(3)=TEMPI
      GO TO 13
   19 IF(RR(3))13,13,20
   20 TEMPR=RR(3)
      TEMPI=RI(3)
      RR(3)=RR(4)
      RI(3)=RI(4)
      RR(4)=TEMPR
      RI(4)=TEMPI
   13 DO 140 I=1,4
  140 CALL POLYVAL (AR,AI,N,RR(I),RI(I),PZR(I),PZI(I),NRLCMP,NPREC)
      DO 137 I=1,4
      IF(ABSF(PZR(I))-1.E-6)135,135,148
  135 IF(ABSF(PZI(I))-1.E-6)137,137,148
  148 L=1
      GO TO 103
  137 CONTINUE
   69 IF(RR(2))70,70,71
   70 IF(RR(3))72,72,99
   71 L=2
      GO TO 103

C REJECT CASES WHERE ONE OR THREE ROOTS HAVE NEGATIVE REAL PART
C TEST THE ACCURACY OF THE ROOTS BY EVALUATING THE POLYNOMIAL

   72 L=3
      GO TO 103
   99 CALL MULT(RR(2),RI(2),RR(2),RI(2),RR2,RI2,K1)
      CALL MULT(RR(1),RI(1),RR(1),RI(1),RR1,RI1,K1)
      ZR1=RR1-RR(1)/C
      ZR2=RR2-RR(2)/C
      ZI1=RI1-RI(1)/C
      ZI2=RI2-RI(2)/C
      CALL DIVD(ZR1,ZI1,G,W,ZSR1,ZSI1,K1)
      CALL DIVD(ZR2,ZI2,G,W,ZSR2,ZSI2,K1)
```

```
      CNR1=ZSR2-1.0/C
      CNI1=ZSI2
      CNR2=1.0/C-ZSR1
      CNI2=-ZSI1
      DENR=ZR2-ZR1
      DENI=ZI2-ZI1
      CALL DIVD(CNR1,CNI1,DENR,DENI,CR1,CI1,K1)
      CALL DIVD(CNR2,CNI2,DENR,DENI,CR2,CI2,K1)
      ER1=EXPF(RR(1)*Y)
      ER2=EXPF(RR(2)*Y)
      CFI1=COSF(RI(1)*Y)
      CFI2=COSF(RI(2)*Y)
      SFI1=SINF(RI(1)*Y)
      SFI2=SINF(RI(2)*Y)
      ERR1=ER1*CFI1
      ERR2=ER2*CFI2
      EII1=ER1*SFI1
      EII2=ER2*SFI2
      CALL MULT(CR1,CI1,ERR1,EII1,VR1,VI1,K1)
      CALL MULT(CR2,CI2,ERR2,EII2,VR2,VI2,K1)
      BDN=G**2+W**2
      BR1=G/BDN
      BI1=-W/BDN
      BR2=1.0/C
      BI2=0
      BBR1=CR1+CR2
      BBI1=CI1+CI2
      CALL MULT(CR1,CI1,ZR1,ZI1,CZR1,CZI1,K1)
      CALL MULT(CR2,CI2,ZR2,ZI2,CZR2,CZI2,K1)
      BBR2=CZR1+CZR2
      BBI2=CZI1+CZI2

C CHECK THE SOLUTIONS OF THE SIMULTANEOUS EQUATIONS

      IF(ABSF(BR1-BBR1)-1.E-6)91,91,94
   91 IF(ABSF(BI1-BBI1)-1.E-6)92,92,94
```

```
   92 IF(ABSF(BR2-BBR2)-1.E-6)93,93,94
   93 IF(ABSF(BI2-BBI2)-1.E-6)96,96,94
   94 L=4

C IF SUBROUTINE POLYRT FAILS,GO BACK TO COMSUB OR VICE VERSA

  103 DO 200 I=1,4
      RI(I)=0.
  200 RR(I)=0.
      MG=MG+2
      IF(MG-3)130,131,104
   96 CALL MULT(ZR1,ZI1,VR1,VI1,UR1,UI1,K1)
      CALL MULT(ZR2,ZI2,VR2,VI2,UR2,UI2,K1)
      GR=1.+G/C
      GI=W/C
      VR=VR1+VR2
      VI=VI1+VI2
      CALL MULT(GR,GI,VR,VI,UR3,UI3,K1)
      UR=-UR1-UR2+UR3
      UI=-UI1-UI2+UI3

C SET VR,VI,UR AND UI EQUAL TO ZERO IF THE CALCULATIONS OF THE ROOTS OR
C CONSTANTS FAIL

      GO TO 109
  104 GO TO(31,32,33,34),L
   31 PRINT 41,N1,M1,MG
   41 FORMAT(/,5X,3HN =I3,5X,3HJ =I3,5X,33H PZR OR PZI IS LARGER THAN 1.
     1 E-4,5X,5HMG = I3)
      GO TO 109
   32 PRINT 42,N1,M1,MG
   42 FORMAT(/,5X,3HN =I3,5X,3HJ =I3,5X,32H ONE ROOT HAS NEGATIVE REAL P
     1ART,5X,5HMG = I3)
      GO TO 109
   33 PRINT 43,N1,M1,MG
   43 FORMAT(/,5X,3HN =I3,5X,3HJ =I3,5X,36H THREE ROOTS HAVE NEGATIVE RE
```

```
   1AL PART,5X,5HMG = I3)
    GO TO 109
 34 PRINT 44,N1,M1,MG
 44 FORMAT(/,5X,3HN =I3,5X,3HJ =I3,5X,42H CONSTANT VECTOR NOT EQUAL TO
   1 CHECK VECTOR,5X,5HMG = I3)
109 VR=0.
    VI=0.
    UR=0.
    UI=0.
110 MG=1
    RETURN
    END

    SUBROUTINE COMSUB(N,MG,AR,AI,XX,YY,IND,XGU,YGU)
   0DIMENSION AR(50),AI(50),DR(50),DI(50),XGU(50),YGU(50),DRU(50),DIU(
   150),ARL(50),AIL(50),ARU(50),AIU(50),IND(50),XX(50),YY(50)
    E1=5.E-6
    E2=1.E-9
    E3=1.E-15
    E4=1.E-100
    E5=1.E-3
 44 IF(N) 97,97,14
 97 STOP 97
 14 NP1=N+1
    AAU=AR(1)
    BBU=AI(1)
    AAL=0.0
    BBL=0.0
    CALL RANMUL(AAU,AAL,AAU,AAL,UU1,LL1)
    CALL RANMUL(BBU,BBL,BBU,BBL,UU2,LL2)
    CALL RANADD(UU1,LL1,UU2,LL2,DNU,DNL)
    DO 17 I=1,N
    ARL(I+1)=0.0
    AIL(I+1)=0.0
    CALL RANMUL(AAU,AAL,AR(I+1),ARL(I+1),UU1,LL1)
    CALL RANMUL(BBU,BBL,AI(I+1),AIL(I+1),UU2,LL2)
```

```
      CALL RANADD(UU1,LL1,UU2,LL2,NUMU,NUML)
      CALL RANDIV(NUMU,NUML,DNU,DNL,ARU(I),ARL(I))
      CALL RANMUL(AAU,AAL,AI(I+1),AIL(I+1),UU1,LL1)
      CALL RANMUL(AR(I+1),ARL(I+1),BBU,BBL,UU2,LL2)
      CALL RANSUB(UU1,LL1,UU2,LL2,NUMU,NUML)
      CALL RANDIV(NUMU,NUML,DNU,DNL,AIU(I),AIL(I))
      DRU(I)=ARU(I)
   17 DIU(I)=AIU(I)
   21 IF(MG)15,23,15
   23 XGU(1)=1.E-3
      YGU(1)=1.E-3
   15 DO 96 NU=1,N
      LIMIT=N-NU+1
      PHU=0.0
      PKU=0.0
   34 XU=XGU(NU)
      XL=0.0
      YU=YGU(NU)
      YL=0.0
   37 DO 36 ITERS=1,50
      AU=1.0
      BU=0.0
      GU=1.0
      DU=0.0
   27 DU1=0.0
      GU1=1.0
      DO 30 K=1,LIMIT
      DU=DU1
      GU=GU1
      AU1=DRU(K)+XU*AU-YU*BU
      BU=DIU(K)+XU*BU+YU*AU
      AU=AU1
      GU1=AU+XU*GU-YU*DU
      DU1=BU+XU*DU+YU*GU
   30 CONTINUE
      CAP=GU*GU+DU*DU
```

```
      PHU=(AU*GU+BU*DU)/CAP
      PKU=(BU*GU-AU*DU)/CAP
   55 ZDEL=PHU*PHU+PKU*PKU
      ZR=(XU*XU+YU*YU)*4.0
      IF(ZR-ZDEL)80,81,81
   80 XU=XU-SIGNF(2.0*XU,PHU)
      YU=YU-SIGNF(2.0*YU,PKU)
      GO TO 36
   81 XU=XU-PHU
      YU=YU-PKU
      AXU=ABSF(XU)
      AYU=ABSF(YU)
      IF(AXU-E3)56,56,71
   71 IF(ABSF(PHU)-E1*AXU)56,56,36
   56 IF(AYU-E3)39,39,72
   72 IF(ABSF(PKU)-E1*AYU)39,39,36
   39 MODE=-1
      GO TO 31
   36 CONTINUE
   62 MODE=0
   31 XL=0.0
      YL=0.0
      DO 1000 ITER=1,50
      AL=0.0
      AU=1.0
      BU=0.0
      GU=1.0
      DU=0.0
      BL=0.0
      GL=0.0
      DL=0.0
      DO 32 K=1,N
      CALL RANMUL(XU,XL,AU,AL,UU1,LL1)
      CALL RANMUL(YU,YL,BU,BL,UU2,LL2)
      CALL RANSUB(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANADD(ARU(K),ARL(K),UU1,LL1,APRU,APRL)
```

```
      CALL RANMUL(XU,XL,BU,BL,UU1,LL1)
      CALL RANMUL(YU,YL,AU,AL,UU2,LL2)
      CALL RANADD(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANADD(AIU(K),AIL(K),UU1,LL1,BU,BL)
      AU=APRU
      AL=APRL
      GU1=GU
      DU1=DU
      GL1=GL
      DL1=DL
      CALL RANMUL(XU,XL,GU,GL,UU1,LL1)
      CALL RANMUL(YU,YL,DU,DL,UU2,LL2)
      CALL RANSUB(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANADD(AU,AL,UU1,LL1,GARU,GARL)
      CALL RANMUL(XU,XL,DU,DL,UU1,LL1)
      CALL RANMUL(YU,YL,GU,GL,UU2,LL2)
      CALL RANADD(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANADD(BU,BL,UU1,LL1,DU,DL)
      GU=GARU
   32 GL=GARL
   33 CALL RANMUL(GU1,GL1,GU1,GL1,UU1,LL1)
      CALL RANMUL(DU1,DL1,DU1,DL1,UU2,LL2)
      CALL RANADD(UU1,LL1,UU2,LL2,CAPU,CAPL)
      CALL RANMUL(AU,AL,GU1,GL1,UU1,LL1)
      CALL RANMUL(BU,BL,DU1,DL1,UU2,LL2)
      CALL RANADD(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANDIV(UU1,LL1,CAPU,CAPL,PHU,PHL)
      CALL RANMUL(BU,BL,GU1,GL1,UU1,LL1)
      CALL RANMUL(AU,AL,DU1,DL1,UU2,LL2)
      CALL RANSUB(UU1,LL1,UU2,LL2,UU1,LL1)
      CALL RANDIV(UU1,LL1,CAPU,CAPL,PKU,PKL)
   35 CALL RANSUB(XU,XL,PHU,PHL,XU,XL)
      CALL RANSUB(YU,YL,PKU,PKL,YU,YL)
      AYU=ABSF(YU)
      AXU=ABSF(XU)
      IF(AXU-E4)73,73,75
```

```
   75 IF(ABSF(PHU)-E2*AXU)73,73,1000
   73 IF(AYU-E4)41,41,74
   74 IF(ABSF(PKU)-E2*AYU)41,41,1000
 1000 CONTINUE
      IF(MODE)42,61,98
   42 IND(NU)=1
      GO TO 45
   41 IND(NU)=0
   45 XX(NU)=XU
      YY(NU)=YU
      NLIMIT=LIMIT-1
      IF(NLIMIT)98,118,82
   82 BRO=1.0
      BIO=0.0
      DO 77 I=1,NLIMIT
      BR1=DRU(I)+XU*BRO-YU*BIO
      BI1=DIU(I)+YU*BRO+XU*BIO
      BRO=BR1
      BIO=BI1
      DRU(I)=BR1
   77 DIU(I)=BI1
      IF(MG)96,79,96
   79 IF(ABSF(YU/XU)-E5)83,78,78
   83 YGU(NU+1)=XU
      GO TO 84
   78 YGU(NU+1)=-YU
   84 XGU(NU+1)=XU
   96 CONTINUE
   61 IND(NU)=-1
  118 RETURN
   98 STOP 98
   99 STOP 99
      END
C
```

```
        MACHINE POLYVAL(A,B,NDEG,ZR,ZI,RR,RI,NR,NP)
        LOC(Z=0,ERP=24)
        CON(H1=6151672065515120B, H2=4746436525614300B)
        LIB(DPFMU,DPFAD,DPSTD,DPDTS)
        RSV(AZR=1,AZI=1,RR=1,RI=1,ANZI=1,TRR=1,TRI=1,TEM=1)
1PVAL   SLJ(*)          SIL4(2EX)       .
1A      SIL2(1EX)       ENI4(*)         .A
2A      SIU3(2EX)       ENI2(*)         .B
3A      SIU5(3EX)       LIL5(*)         .B5=NDEG
4A      SIU1(1EX)       LDA(*)          .ZR
5A      STA(ZR)         LDA(*)          .ZI
6A      STA(ZI)         ENI(*)          .ER
7A      ENQ5(-1B)       ENI(*)          .EI
        QJP3(1ERR)      LIL3(*)         .NRC, OREAL, NOT ZERO COMPLEX
        ENI(*)          LDA(*)          .NPRECISION
        INA(-1B)        AJP(1LOOP)      .
        INA(-1B)        AJP(2LOOP)      .
1ERR    ENA(H1)         SLJ4(ERP)       .
1LOOP   IJP3(L+1)       SLJ(1REAL)      .JUMP IF REAL POLY
1COM    SIL2(1SG1)      SIU4(1SG2)      .PRESET FOR LOOP
        LDA4(Z+1)       STA(RR)         .RR = A(1)
        LDA2(Z+1)       STA(RI)         .RI = B(1)
        ENI4(2)         INI5(-1B)       .
1SGL    LDA(ZR)         FMU(RR)         .
        STA(TRR)        LAC(ZI)         .
        FMU(RI)         FAD(TRR)        .
        STA(TRR)        LDA(ZR)         .
        FMU(RI)         STA(TRI)        .
        LDA(ZI)         FMU(RR)         .
1SG1    FAD(TRI)        FAD4(*)         .
        STA(RI)         LDA(TRR)        .
1SG2    FAD4(*)         STA(RR)         .
        INI4(1)         IJP5(1SGL)      .
        STA7(6A)        LDA(RI)         .
        STA7(7A)        SLJ(1EX)        .
1REAL   INI5(-1B)       LDA4(Z+1)       .
```

```
        FMU(ZR)      FAD4(Z+2)    .
        INI4(1)      IJP5(L-1)    .
9FX     STA7(6A)     ENA(0)       .
8FX     STA7(7A)     SLJ(1EX)     .
2LOOP   IJP3(L+1)    SLJ(2REAL)   .
2COM    SIU4(1ACON)  LDA4(Z+1)    .
        ENI1(RR)     SLJ4(DPSTD)  .
        SIU2(2ABCO)  LDA2(Z+1)    .
        ENI1(RI)     SLJ4(DPSTD)  .
        LDA(ZR)      ENI1(AZR)    .
        SLJ4(DPSTD)               .
        LDA(ZI)      ENI1(AZI)    .
        SLJ4(DPSTD)               .
        LAC(ZI)      ENI1(ANZI)   .
        SLJ4(DPSTD)               .
        ENI4(2)      INI5(-1B)    .
2DBL    ENI1(AZR)    ENI2(RR)     .
        ENI3(TEM)    SLJ4(DPFMU)  .
        ENI1(ANZI)   ENI2(RI)     .
        ENI3(TRR)    SLJ4(DPFMU)  .
        ENI1(TRR)    ENI2(TEM)    .
        ENI3(TRR)    SLJ4(DPFAD)  .
        ENI1(AZR)    ENI2(RI)     .
        ENI3(TEM)    SLJ4(DPFMU)  .
        ENI1(AZI)    ENI2(RR)     .
        ENI3(TRI)    SLJ4(DPFMU)  .
        ENI1(TRI)    ENI2(TEM)    .
        ENI3(TRI)    SLJ4(DPFAD)  .
1ACON   LDA4(*)      ENI1(TEM)    .
        SLJ4(DPSTD)               .
        ENI1(TRR)    ENI2(TEM)    .
        ENI3(RR)     SLJ4(DPFAD)  .
2ABCO   LDA4(*)      ENI1(TEM)    .
        SLJ4(DPSTD)               .
        ENI1(TRI)    ENI2(TEM)    .
        ENI3(RI)     SLJ4(DPFAD)  .
```

```
      INI4(1)        IJP5(2DBL)      .
      ENI1(RR)       SLJ4(DPDTS)     .
      STQ7(6A)       FNI1(RI)        .
      SLJ4(DPDTS)                    .
      LLS(48)        SLJ(8EX)        .
2REAL LDA4(Z+1)      ENI1(R)         .
      ENI3(R)        SLJ4(DPSTD)     .
      LDA(ZR)        ENI1(ZR)        .
      INI5(-1B)      SLJ4(DPSTD)     .
1DBL  FNI2(R)        SLJ4(DPFMU)     .
      LDA4(Z+2)      FNI1(TFM)       .
      FNI2(R)        SLJ4(DPSTD)     .
      INI4(1)        SLJ4(DPFAD)     .
      FNI1(ZR)       IJP5(1DBL)      .
      ENI1(R)        SLJ4(DPDTS)     .
      LLS(48)        SLJ(9EX)        .
1EX   FNI1(*)        ENI2(*)         .
2FX   FNI3(*)        FNI4(*)         .
3EX   FNI5(*)        SLJ(1PVAL)      .
      END
      MACHINE DPDTS
C
C     DOUBLE PRECISION AT A TO SINGLE PRECISION FLOATING NUMBER AT C
C     IF EXPONENT OVERFLOW, C SET TO LARGEST POSITIVE NUMBER
C     APPROXIMATE TIME FOR DPDTS  107 MICROSECONDS + 28 SETUP
C
      LOC(Z=0)
      CON(CON0=3777777777777776B)
1DTS  SLJ(*)      LDA1(Z+1)
      SAU(L+1)         LDQ1(Z)           .
      ENA(*)           AJP2(L+2)         .BIAS EXPONENT
      INA(-1)                            .
      INA(2000B)       AJP3(1ZRO)        .TEST IF ZERO
      THS(4000B)       SLJ(1OVER)        .TEST FOR OVERFLOW
      QJP2(L+1)        SCM(-0B)          .
      QLS(1)           LRS(12)           .PACK EXPONENT AND MANTISSA
```

```
      SLJ(1DTS)                                .
1OVER LDQ(CONO)       SLJ(1DTS)
1ZRO  ENQ(0)      SLJ(1DTS)
      END
      MACHINE DPSTD
C
C     SINGLE PRECISION FLOATING NUMBER AT A TO DOUBLE PRECISION AT C
C     APPROXIMATE TIME FOR DPSTD 106 MICROSECONDS + 28 SETUP
C
      LOC(Z=0)
      CON(CONO=4000000000000000B)
1STD  SLJ(*)      AJP(1ZRO)
      AJP3(3STD)        ENQ(0)          .TEST IF A NEGATIVE
      LLS(12)           ARS(1)          .
      SCL(CONO)         STA1(Z)         .STORE MANTISSA OF C
      LDL(-0B)          INA(-2000B)     .UNPACK EXPONENT AND STORE
2STD  AJP2(L+1)         INA(1)          .
      STQ1(Z+1)         SAL1(Z+1)       .
1EXIT SLJ(1STD)
3STD  ENQ(-0)           LLS(12)         .
      ARS(1)            SST(CONO)       .
      STA1(Z)           ENA(0)          .STORE MANTISSA OF C
      SBL(-0B)          INA(-2000B)     .UNPACK EXPONENT AND STORE
      SLJ(2STD)                         .
1ZRO  STA1(Z)      ENI(*)
      ENA(24000B)       ALS(1)          .
      STA1(Z+1)         SLJ(1EXIT)      .
      END
      MACHINE DPFMU
C
C     D(A)*D(B) TO C, 0 TO Q1604
C     IF A UPPER TIMES B UPPER ZERO, THEN PRODUCT C IS ZERO
C     IF EXPONENT OVERFLOW, THEN C SET TO LARGE POSITIVE VALUE, -1 TO Q1
C     APPROXIMATE TIME FOR DPFMU  384 MICROSECONDS + 38 SETUP
C
      LOC(Z=0)
```

```
       CON(CON0=70000000000007777B)
1M     SLJ(*)        LDA1(Z)
1AC    MUF2(Z)            ENI(*)             .ADDRESS OF C
       AJP(1ZRO)          STA(UPPER)         .TEST IF PRODUCT AU*BU IS ZERO
       ALS(1)             LRS(1)             .
       QRS(2)             STQ(LOWER)         .STORE AU*BU IN UPPER AND LOWER
       LDA1(Z+1)          SAU(1EXP)          .
       ARS(3)             SST(CON0)          .
       SSK1(Z)            SCL(CON0)          .
       MUF2(Z)            RAD(LOWER)         .AL*BU + LOWER TO LOWER
       LDA2(Z+1)          SAL(1EXP)          .
       ARS(3)             SST(CON0)          .
       SSK2(Z)            SCL(CON0)          .
       MUF1(Z)            ADD(LOWER)         .AU*BL + LOWER
       LRS(45)            ADD(UPPER)         .
       SCQ2(96)       STA3(Z)
       STQ3(Z+1)      ENA2(-96)
1EXP   INA(*)        INA(*)
       SAL3(Z+1)         INA(8000)
       AJP2(1M)
1ZRO   ENQ(0)        STQ3(Z)
       ENA(24000B)        ALS(1)             .
       STA3(Z+1)         SLJ(1M)
       END
       MACHINE DPFAD
C
C      D(A)+D(B) TO C, 0 TO Q1604
C      IF EXPONENT OVERFLOW, THEN C SET TO LARGE POSITIVE VALUE,-1 TO Q16
C      APPROXIMATE TIME FOR DPFAD 361 MICROSECONDS + 38 SETUP
C
       LOC(Z=0)
       CON(CON0=17777777777777777B,CON1=20000000000000000B)
       CON(CON3=6000 0000 0000 0000B)
1ADD   SLJ(*)        LDA1(Z+1)    .
1AC    SAU(1EXP)          ENI(*)             .ADDRESS OF C
       LAC2(Z+1)          SAL(1EXP)          .
```

```
1EXP  ENA(*)          INA(*)          .COMPARE EXPONFNTS
      AJP3(1BGR)      INA(13)         .A GREATER OR FQUAL IN EXP
      THS(94)         SLJ(1BNS)       .TEST SIGNIFICANCF OF B WITH A
      SAL(L+1)        LDA2(Z)         .
      LDQ2(Z+1)       LRS(*)          .POSITION B
      QRS(2)          STA(BU)         .
      STQ(BL)         LDA1(Z)         .POSITION A
      LDQ1(Z+1)       LRS(13)         .
      LIU1(1EXP)      SIU1(2EXP)      .
2ADD  SCL(CON3)       QRS(2)          .
      STQ(AU)         LDQ(CON0)       .LOAD MASK
      ENI1(0)         ADL(BU)         .ADD
      THS(CON1)       ENI1(1)         .TEST FOR END AROUND CARRY
      STL(BU)         ENA1(0)         .
      ADL(BL)         ADL(AU)         .
      LRS(46)         ADD(BU)         .
      THS(CON1)       SLJ(4ADD)       .TFST FOR SFCOND CARRY
3ADD  LLS(2)          LRS(4)          .FXTFND SIGN BITS
      SCQ2(95)        AJP(1ZRO)       .NORMALIZF AND STORF IN C
      STA3(Z)       STQ3(Z+1)
2FXP  ENA(*)      INA2(-80)    . DFTERMINT AND STORE EXPONFNT
      SAL3(Z+1)      INA(8000)  .
      AJP2(1ADD)                .
1ZRO  ENQ(0)      STQ3(Z)
      FNA(24000B)     ALS(1)          .
      STA3(Z+1)      SLJ(1ADD)
4ADD  LLS(48)         INA(4)          .SFCOND END AROUND CARRY
      LLS(48)         SLJ(3ADD)       .
1ANS  SIL2(1AC)      LIL1(1AC)
1BNS  LDA1(Z)         LDQ1(Z+1)       .B NOT SIGNIFICANT W.R. TO A
      STA3(Z)     STQ3(Z+1)
      SLJ(1ADD)
1BGR  SCM(-0B)        INA(13)         .B GREATER IN FXPONENT
      THS(94)         SLJ(1ANS)       .TEST SIGNIFICANCF OF B WITH A
      SAL(L+1)        LDA1(Z)         .
      LDQ1(Z+1)       LRS(*)          .POSITION A
```

```
QRS(2)          STA(BU)        .
STQ(BL)         LAC(1FXP)      .
SAU(2EXP)       LDA2(Z)        .POSITION B
LDQ2(Z+1)       LRS(13)        .
SLJ(2ADD)                      .
END
END
```

```
..JOB0115F,HIEP SCHUMAN
      PROGRAM SCHUMAN

C ANALYTICAL SOLUTION OF SCHUMANN,S PROBLEM
C LONGITUDINAL CONDUCTION IN FLUID AND SOLID PHASE ARE NEGLECTED
C DIMENSIONLESS TEMPERATURE V.S DIMENSIONLESS DISTANCE
C DIMENSIONLESS DISTANCE FROM 0 TO INFINITY

C A IS RATIO OF THERMAL DIFFUSIVITIES
C B IS RATIO OF THERMAL CONDUCTIVITIES
C C IS DIMENSIONLESS PARAMETER LAMBDA
C T IS DIMENSIONLESS TIME
C X IS DIMENSIONLESS DISTANCE
C M IS RUN NUMBER
C SET M = 0 ON LAST DATA CARD

      COMMON E,P,Q
      INIT=-1
  100 READ 101,A,B,C,T,M
  101 FORMAT(4F10.5,I3)
      IF(M)200,200,201
  201 PRINT 102,A,B,C,T
  102 FORMAT(1H1,///,20X,3HA =F15.5,5X,3HB =F15.5,5X,3HC =F15.5,5X,
     13HT =F15.5,///)
      PRINT 103
  103 FORMAT(40X,1HY,18X,1HV,20X,1HU,///)
      Y=-10.
  400 Y=Y+10.
      X0=0.0
      XL=T-Y
      REL=1.E-8
      NP=5
      E=1.
      P=A*B*C*C*Y
      Q=A*B*C
```

```
C CALCULATION OF THE INTEGRAL BY GAUSS QUADRATURE

      CALL GAUSSN(INIT,XO,XL,G,REL,NP)
      V=EXPF(-C*Y)*(1.0+SQRTF(P)*G)
      F=0.
      REL=1.E-8
      NP=5
      CALL GAUSSN(INIT,XO,XL,G,REL,NP)
      U=Q*EXPF(-C*Y)*G
      PRINT 300,Y,V,U
  300 FORMAT(/,34X,E10.5,2E21.9)
      IF(V-1.E-4)100,400,400
  200 STOP
      END

      SUBROUTINE GAUSSN(INIT,XO,XL,Y,REL,NP)
C     TO CONVERT FROM GAUSS16 TO GAUSSN, CHANGE THE CARDS WITH
C     COMMENTS, WHERE N = ORDER OF FORMULA.
      DIMENSION AA(16),HH(16),YBAR(10),BYB(10)
      COMMON F,P,Q
      IF(INIT)1,1,2
 1    INIT = -INIT
      AA(1) = -.98940093499
      AA(2) = -.94457502307
      AA(3) = -.86563120239
      AA(4) = -.75540440836
      AA(5) = -.61787624440
      AA(6) = -.45801677766
      AA(7) = -.28160355078
      AA(8) = -.95012509838E-01
      AA(9) = -AA(8)
      AA(10) = -AA(7)
      AA(11) = -AA(6)
      AA(12) = -AA(5)
      AA(13) = -AA(4)
      AA(14) = -AA(3)
```

```
      AA(15) = -AA(2)
      AA(16) = -AA(1)
      HH(1) = .27152459412E-01
      HH(2) = .62253523939E-01
      HH(3) = .95158511682E-01
      HH(4) = .12462897126
      HH(5) = .14959598882
      HH(6) = .16915651940
      HH(7) = .18260341504
      HH(8) = .18945061046
      HH(9) = HH(8)
      HH(10) = HH(7)
      HH(11) = HH(6)
      HH(12) = HH(5)
      HH(13) = HH(4)
      HH(14) = HH(3)
      HH(15) = HH(2)
      HH(16) = HH(1)
      NG = 16
2     Y = 0.
      XLGTH = XL-XO
      IF(XLGTH)201,105,201
201   NPP = NP
      DO 103 K = 1,10
      Y = 0.
      FNP = NP
      DO 200 L = 1,NP
      AREA = 0.
      AL = L
      X1PX2 = (2.*AL-1.)*XLGTH/FNP + 2.*XO
      X2MX1 = XLGTH/FNP
      DO 100 J = 1,NG
      X = (X1PX2 + AA(J) * X2MX1)/2.
      CALL FOFX(X,FX)
100   AREA = AREA + HH(J)*FX
      Y = Y + AREA
```

```
200   CONTINUE
      Y = XLGTH/(2.*FNP) * Y
      YBAR(K) = Y
      IF(K-1)104,104,144
144   BYB(K-1) = ABSF(YBAR(K-1) - Y)
      IF(BYB(K-1) - REL*ABSF(Y))105,105,104
104   NP = 2*NP
103   CONTINUE
      DO 108 L = 1,10
      REL = 2.*REL
      DO 107 K = 2,10
      IF(BYB(K-1) - REL*ABSF(YBAR(K)))106,106,107
107   CONTINUE
108   CONTINUE
      K = 10
106   NP = (2**(K-1)) * NPP
      Y = YBAR(K)
105   RETURN
      END

      SUBROUTINE FOFX(T,FT)
      COMMON E,P,Q
      W=2.*SQRTF(P*T)
      CALL BESSELL(E,W,Z)
      IF(E)5,10,5
    5 FT=EXPF(-Q*T)*Z/SQRTF(T)
      GO TO 15
   10 FT=EXPF(-Q*T)*Z
   15 RETURN
      END

      SUBROUTINE BESSELL(A,X,Z)
      DIMENSION C(9)
      IF(A)2,1,2
    1 IF(X)2,4,2
    2 H=0.
```

```
      B=X*X/4.
      C(1)=1.
      C(2)=-.577191652
      C(3)=0.988205891
      C(4)=-.897056937
      C(5)=0.918206857
      C(6)= -.756704078
      C(7)=0.482199394
      C(8)=-.193527818
      C(9)=0.035868343
      D=0.
      DO 3 I=1,9
    3 D=D*A+C(10-I)
      X2=(X/2.)**A/D
      Z=X2
   10 W=Z
      H=H+1.
      HA=H+A
      X1=X2
      X2=B*X1/(H*HA)
      Z=Z+X2
      IF(Z-W-1.E-11)12,10,10
    4 Z=1.
   12 RETURN
      END
      END
```

```
..JOB0115F,HIEP,JENKINS
      PROGRAM JENKINS

C ANALYTICAL SOLUTION OF JENKINS AND ARONOFSKY,S PROBLEM
C FLUID AND SOLID TEMPERATURES ASSUMED TO BE EQUAL(HA=INFINITY)

C DIMENSIONLESS TEMPERATURE V.S DIMENSIONLESS DISTANCE
C DIMENSIONLESS DISTANCE FROM 0 TO INFINITY
C A IS RATIO OF THERMAL DIFFUSIVITIES
C B IS RATIO OF THERMAL CONDUCTIVITIES
C C IS DIMENSIONLESS PARAMETER LAMBDA
C T IS DIMENSIONLESS TIME
C Y IS DIMENSIONLESS DISTANCE
C M IS RUN NUMBER
C SET M = 0 ON LAST DATA CARD

      DIMENSION SF(500)
  100 READ 101,A,B,C,T,M
  101 FORMAT(4F10.5,I3)
      IF(M)105,300,105
  105 PRINT 102,A,B,C,T
  102 FORMAT(1H1,///,20X,3HA =F15.5,5X,3HB =F15.5,5X,3HC =F15.5,5X,
     13HT =F15.5,///)
      PRINT 104
  104 FORMAT(///,10X,9H DISTANCE,15X,3HERC,15X,2HE2,15X,1HV,//)
      X=-5.
  200 X=X+5.
      A1=(1.+1./B)*C
      C1=1.+1./(A*B)
      P=A1/C1
      Q=1./(4.*A1*C1)
      Z=X/(2.*SQRTF(P*T))
      W=SQRTF(Q*T)
      Y1=Z-W
      Y2=Z+W
```

```
      IF(Y1)205,210,210
  205 Y1=-Y1
      E1=ERFN(Y1)
      E1=-E1
      GO TO 220
  210 E1=ERFN(Y1)
  220 FRC=1.-E1
      F=1.
      SUM=0.

C SERIES EXPANSION OF THE COMPLEMENTARY ERROR FUNCTION

      DO 14 N=1,500
      G=N
      IF(N-1)12,13,12
   12 F=-F*(2.*G-3.)
   13 SE(N)=F/(2.**(N-1)*Y2**(2*N-1))
      SUM=SUM+SE(N)
      S1=ABSF(SE(N-1))
      S2=ABSF(SE(N))
      IF(ABSF(S1-S2)-1.E-6)15,15,14
   14 CONTINUE
   15 F2=EXPF(2.*X*SQRTF(Q/P)-Y2*Y2)*SUM/SQRTF(3.1415926536)
      V=0.5*(FRC+F2)
      PRINT 250,X,ERC,F2,V
  250 FORMAT(/,5X,F10.5,5X,3E20.8)
      IF(V-1.E-4)100,200,200
  300 STOP
      END
```

```
 MACHINE ERFN(1A)                                                      000
 LOC(ERP=24).                                                          000
 CON(K2=200077777777776B)                                              000
 HOL(H1=NEG. ARG, H2= IN ERFN).                                        000
 CON(K1=0)                                                             000
   CON(CNTRPT6=20034114631463 15B,  CNTRPT5=20025714631463 15B,        000
*       CNTRPT4=2002434631463146B,  CNTRPT3=2001640000000000B,         000
*       CNTRPT2=2000731463146314B,  CNTRPT1=1776463146314631B)         000
   CON(RNG6=2003447534121727B,  RNG5=2002700000000000B,                000
*       RNG4=2002471463146315B,  RNG3=2002400000000000B,               000
*       RNG2=2001500000000000B,  RNG1=2000500000000000B)               000
 CON(CN6 1=2000777777777321B,  CN5 1=2000777700521161B,                000
*     CN4 1=2000777117015156B,  CN3 1=2000764755116661B,               000
*     CN2 1=2000636230646226B,  CN1 1=1776520407042013B)               000
 CON(CN6 2=1747501510506000B,  CN5 2=1763611161501200B,                000
*     CN4 2=1771405574523127B,  CN3 2=1774511472436074B,               000
*     CN2 2=1776753056373171B,  CN1 2=2001410001334066B)               000
 CON(CN6 3=6026265717617364B,  CN5 3=6012335774572405B,                000
*     CN4 3=6005334636712502B,  CN3 3=6002364140335114B,               000
*     CN2 3=6001071573311160B,  CN1 3=6001303144541667B)               000
 CON(CN6 4=1752675336125607B,  CN5 4=1766414617203400B,                000
*     CN4 4=1772604235227060B,  CN3 4=1774726306205223B,               000
*     CN2 4=1774721557322562B,  CN1 4=6001337266624623B)               000
 CON(CN6 5=6024073715225420B,  CN5 5=6011243755366647B,                000
*     CN4 5=6005261045717762B,  CN3 5=6004150545227235B,               000
*     CN2 5=1774606222561656B,  CN1 5=1775451625712515B)               000
 CON(CN6 6=1754517311101456B,  CN5 6=1766512141230047B,                000
*     CN4 6=1771553155776311B,  CN3 6=6010350067147520B,               000
*     CN2 6=6003343706407115B,  CN1 6=1774422707067741B)               000
 CON(CN6 7=6023247745343644B,  CN5 7=6012056422714724B,                000
*     CN4 7=6010224257737547B,  CN3 7=1771730621234621B,               000
*     CN2 7=6007367165234377B,  CN1 7=6004213523602274B)               000
 CON(CN6 8=1754547030154766B,  CN5 8=1764722756771323B,                000
```

```
      *     CN4 8=601204776455237 6B,  CN3 8=600715315100201 3B,            000
      *     CN2 8=177243734000155 0B,  CN1 8=600616556551133 4B)            000
       CON(CN6 9=602305661135154 3B,  CN5 9=601437406311343 2B,            000
      *     CN4 9=176645173524521 2B,  CN3 9=601222523076345 2B,            000
      *     CN2 9=601017333560644 5B,  CN1 9=177151632755140 6B)            000
       CON(CN610=175444650234463 2B,  CN510=175247551266715 4B,            000
      *     CN410=601421463301327 3B,  CN310=176654425070477 2B,            000
      *     CN210=601026264444624 7B,  CN110=176654433067750 2B)            000
1OUT  SLJ(N)          SIU1(1END)        .EXIT/ENTRANCE                     000
1A    ENI1(6)         LDA(N)            .ARGUMENT ADDRESS                  000
      AJP3(1ERR)    AJP(1END).                                             000
      THS1(RNG6)      SLJ(3A)           .FIND INTERVAL                     000
      FSB1(CNTRPT6)   STA(T1)           .SUBTRACT CENTER POINT             000
      FMU1(CN610)   FAD1(CN69).                                            000
      FMU(T1)      FAD1(CN68).                                             000
      FMU(T1)      FAD1(CN67).                                             000
      FMU(T1)      FAD1(CN66).                                             000
      FMU(T1)      FAD1(CN65).                                             000
      FMU(T1)      FAD1(CN64).                                             000
      FMU(T1)      FAD1(CN63).                                             000
      FMU(T1)      FAD1(CN62).                                             000
      FMU(T1)      FAD1(CN61).                                             000
1END  ENI1(N)         SLJ(1OUT)         .RESTORE B1,EXIT                   000
3A    LDA(K2)         SLJ(1END)         .                                  000
1ERR  ENA(H1)       SLJ4(ERP).                                             000
      END                                                                  000
      END
```